E-Infrastructures
H2020- INFRAEDI-2018-2020

INFRAEDI-01-2018: Pan-European High Performance Computing infrastructure and services (PRACE)

PRACE-6IP

PRACE Sixth Implementation Phase Project

Grant Agreement Number: INFRAEDI-823767

D7.2

Final Report on Results from Projects Supported by Applications Enabling and Porting Services
Final

Version: 1.0
Author(s): Gabriel Hautreux, CINES; Chris Johnson, EPCC; Sebastian Lührs, JUELICH
Date: 07.12.2021

Project and Deliverable Information Sheet

PRACE Project	**Project Ref. №: INFRAEDI-823767**	
	Project Title: PRACE Sixth Implementation Phase Project	
	Project Web Site: https://www.prace-ri.eu/about/ip-projects/	
	Deliverable ID: D7.2	
	Deliverable Nature: Report	
	Dissemination Level: PU*	**Contractual Date of Delivery:** 31 / December / 2021
		Actual Date of Delivery: 20 / December / 2021
	EC Project Officer: Leonardo Flores Añover	

* - The dissemination level are indicated as follows: **PU** – Public, **CO** – Confidential, only for members of the consortium (including the Commission Services) **CL** – Classified, as referred to in Commission Decision 2005/444/EC.

Document Control Sheet

Document	**Title: Final Report on Results from Projects Supported by Applications Enabling and Porting Services**	
	ID: D7.2	
	Version: 1.0	**Status:** ***Final***
	Available at: https://www.prace-ri.eu/about/ip-projects/	
	Software Tool: Microsoft Word 2016	
	File(s): D7.2.docx	
Authorship	**Written by:**	Gabriel Hautreux, CINES; Chris Johnson, EPCC; Sebastian Lührs, JUELICH
	Contributors:	Aitor Gonzalez-Agirre, BSC Anastasiia Shamakina, HLRS Andreas Hadjigeorgiou, CaSToRC Andrew Emerson, CINECA Andrew Sunderland, STFC Anna Mack, HLRS Belgin Övet, UHEM Bernhard Semlitsch, TU Wien Bertrand Cirou, CINES Camille Parisel, IDRIS Carlos Fernandez Sanchez, CESGA Claudia Blaas-Schenner, TU Wien Constantia Alexandrou, CaSToRC Cristian Morales, BSC Daniel Ward, STFC Eleanor Broadway, EPCC Engelbert Tijskens, UANTWERPEN Eric Pascolo, CINECA Eric Verschuur, TU Delft Frédéric Pinel, U Luxembourg David Henty, EPCC

		Guillaume Houzeaux, BSC Harald Grill, TU Wien Isabelle Dupays, IDRIS Jacob Finkenrath, CaSToRC Janez Povh, UL FME Jean-Christophe Pénalva, CINES Jenny Andrea Amundsen, Sigma2 Jing Gong, KTH João Cardoso, UnivPorto Karim Hasnaoui, IDRIS Kim Serradell, BSC Kyriakos Hadjiyiannakou, CaSToRC Laurent Leger, IDRIS Lilit Axner, KTH Lukas Demovic, CCSAS Mario Acosta, BSC Mark Abraham, ENCCS Marta Villegas, BSC Massimiliano Guarrasi, CINECA Matej Spetko, IT4I Maxime Mogé, SURF Michaela Barth, KTH Michal Pitonak, CCSAS Mikhail Davydenko, TU Delft Neelofer Banglawala, EPCC Olivier Coulaud, INRIA Pavel Tomšič, UL FME Pedro Ojeda-May, SNIC-UU Ricard Borrell, BSC Sameed Hayat, HLRS Soner Steiner, TU Wien Stefan Becuwe, UANTWERPEN Stéphane Lanteri, INRIA Thibaut Very, IDRIS Tomas Brzobohaty, VSB-TUO Tomas Karasek, VSB-TUO Wei Zangh, SNIC-UU Xu Guo, EPCC
	Reviewed by:	Lukasz Dutka, Cyfronet; Dirk Brömmel, JUELICH
	Approved by:	MB/TB

Document Status Sheet

Version	Date	Status	Comments
0.1	03/November/2021	Draft	Initial document setup
0.2	19/November/2021	Draft	PA sections
0.3	20/November/2021	Draft	Integration of SHAPE and DECI sections
0.4	21/November/2021	Draft	Integration of Enhancing the HLST sections
0.5	22/November/2021	Draft	Small fixes
0.6	01/December/2021	Draft	Including reviewer comments
0.7	03/December/2021	Draft	DECI/SHAPE sections updated
0.8	06/December/2021	Draft	HLST sections updated
1.0	07/December/2021	Final	Including reviewer comments

Document Keywords

Keywords:	PRACE, HPC, Research Infrastructure, Preparatory Access, SHAPE, HLST

Table of Contents

List of Figures

List of Tables

References and Applicable Documents

[1] http://www.prace-ri.eu

[2] https://prace-ri.eu/hpc-access/calls-for-proposals/

[3] https://prace-ri.eu/hpc-access/preparatory-access/

[4] PRACE-6IP Deliverable 7.1, "Periodic Report on Results from Projects Supported by Applications Enabling and Porting Services", https://prace-ri.eu/wp-content/uploads/PRACE6IP-D7.1.pdf

[5] PRACE white papers, https://prace-ri.eu/training-support/technical-documentation/white-papers/

[6] Delft3D FM Suite website, https://www.deltares.nl/en/software/delft3d-flexible-mesh-suite

[7] Kernkamp, H. W. J., van Dam, A., Stelling, G. S., de Goede, E. D.: Efficient scheme for the shallow water equations on unstructured grids with application to the Continental Shelf. In: Ocean Dynamics 61(8), 1175 - 1188 (2011)

[8] https://www.arm.com/products/development-tools/server-and-hpc/forge/map

[9] https://tools.bsc.es/paraver

[10] https://www.vi-hps.org/projects/score-p/

[11] https://www.intel.com/content/www/us/en/developer/tools/oneapi/vtune-profiler.html

[12] https://phantomsph.bitbucket.io/

[13] PRACE Whitepaper 298, "OpenMP optimization of the eXtended Discrete Element Method (XDEM)", https://prace-ri.eu/wp-content/uploads/WP298.pdf

[14] https://micc.readthedocs.io/en/latest/

[15] PRACE Whitepaper 308, "Elimination of I/O in Optimization Iterations using OpenFOAM Solvers", https://prace-ri.eu/wp-content/uploads/WP308-Elimination-of-IO-in-Optimization-Iterations-using-OpenFOAM.pdf

[16] http://nek5000.mcs.anl.gov

[17] https://libocca.org

[18] Introducing Application Performance Snapshot, https://www.intel.com/content/www/us/en/develop/documentation/application-snapshot-user-guide/top/introducing-application-performance-snapshot.html

[19] A. Sarje, S. Song, D. Jacobsen et al. Parallel Performance Optimizations on Unstructured Mesh-based Simulations. Procedia Computer Science 51, 2016-2025, 2015.

[20] Sidorenko, D.,et. al., Towards multi-resolution global climate modeling with ECHAM6–FESOM. Part I: model formulation and mean climate. *Climate Dynamics*, 44(3-4), 757-780. (2015)

[21] Offermans, N. et.al. "On the Strong Scaling of the Spectral Element Solver Nek5000 on Petascale Systems", In: Proceedings of the 2016 Exascale Applications and Software Conference (EASC2016): April 25-29 2016, Stockholm, Sweden

[22] PRACE-5IP Deliverable 7.2, https://prace-ri.eu/wp-content/uploads/5IP-D7.2.pdf

[23] PRACE-5IP Deliverable 7.1, "Periodic Report on Applications Enabling Services", http://www.prace-ri.eu/IMG/pdf/D7.1_5ip.pdf

[24] PRACE-3IP Deliverable 5.3.3, "Report on the SHAPE Implementation", http://www.prace-ri.eu/IMG/pdf/D5.3.3_3ip.pdf

[25] PRACE white papers – SHAPE Projects, https://prace-ri.eu/training-support/technical-documentation/white-papers/shape-white-papers/

[26] PRACE SHAPE Success Stories https://prace-ri.eu/category/success-stories/shape-success-stories/

[27] How to Deploy Real-Time Text-to-Speech Applications on GPUs Using TensorRT, https://developer.nvidia.com/blog/how-to-deploy-real-time-text-to-speech-applications-on-gpus-using-tensorrt/

[28] NEMO (Nucleus for European Modelling of the Ocean) Github repository, https://github.com/NVIDIA/NeMo

[29] Theoretical and Computational Biophysics Group, NAMD scalable molecular dynamics https://www.ks.uiuc.edu/Research/namd/

[30] NEMO Consortium, NEMO ocean modelling framework https://www.nemo-ocean.eu/

[31] EPCC, ARCHER2 hardware description https://www.archer2.ac.uk/about/hardware.html

[32] CEA, TGCC Joliot Curie hardware description http://www-hpc.cea.fr/en/complexe/tgcc-JoliotCurie.htm

[33] PRACE UEABS, Unified European Applications Benchmark Suite, https://repository.prace-ri.eu/git/UEABS/ueabs/-/tree/r2.2-dev/

[34] Theoretical and Computational Biophysics Group, NAMD wiki https://www.ks.uiuc.edu/Research/namd/wiki/

[35] NEMO Consortium, NEMO documentation https://forge.ipsl.jussieu.fr/nemo/chrome/site/doc/NEMO/guide/html/guide.html

[36] Bonomi, E., et. al. (1998). Phase-shift plus interpolation: a scheme for high-performance echo-reconstructive imaging. Computers in Physics, 12:126–132.

[37] Berkhout, A. J. (2014a). Review paper: An outlook on the future of seismic imaging, Part I: forward and reverse modelling. Geoph. Prosp., 62(5):911–930.

[38] Berkhout, A. J. (2014b). Review paper: An outlook on the future of seismic imaging, Part II: Full-wavefield migration. Geoph. Prosp., 62(5):931–949.

[39] Davydenko, M. and Verschuur, D. J. (2017). Full-wavefield migration: using surface and internal multiples in imaging. Geophysical Prospecting, 65(1):7–21.

[40] Gazdag, J. and Sguazzero, P. (1984). Migration of seismic data by phase shift plus interpolation. GEOPHYSICS, 49(2):124–131.

[41] https://scorepci.pages.jsc.fz-juelich.de/scorep-pipelines/docs/scorep-6.0/html/index.html

[42] https://ambermd.org/

List of Acronyms and Abbreviations

aisbl	Association International Sans But Lucratif (legal form of the PRACE-RI)
BCO	Benchmark Code Owner
CoE	Center of Excellence
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture (NVIDIA)
DARPA	Defense Advanced Research Projects Agency
DEISA	Distributed European Infrastructure for Supercomputing Applications EU project by leading national HPC centres
DoA	Description of Action (formerly known as DoW)
EC	European Commission
EESI	European Exascale Software Initiative
EoI	Expression of Interest
ESFRI	European Strategy Forum on Research Infrastructures
GB	Giga (= 2^{30} ~ 10^9) Bytes (= 8 bits), also GByte
Gb/s	Giga (= 10^9) bits per second, also Gbit/s
GB/s	Giga (= 10^9) Bytes (= 8 bits) per second, also GByte/s
GÉANT	Collaboration between National Research and Education Networks to build a multi-gigabit pan-European network. The current EC-funded project as of 2015 is GN4.
GFlop/s	Giga (= 10^9) Floating point operations (usually in 64-bit, i.e. DP) per second, also GF/s
GHz	Giga (= 10^9) Hertz, frequency =10^9 periods or clock cycles per second
GPU	Graphic Processing Unit
HET	High Performance Computing in Europe Taskforce. Taskforce by representatives from European HPC community to shape the European HPC Research Infrastructure. Produced the scientific case and valuable groundwork for the PRACE project.
HMM	Hidden Markov Model
HPC	High Performance Computing; Computing at a high performance level at any given time; often used synonym with Supercomputing
HPL	High Performance LINPACK
ISC	International Supercomputing Conference; European equivalent to the US based SCxx conference. Held annually in Germany.
KB	Kilo (= 2^{10} ~10^3) Bytes (= 8 bits), also KByte
LINPACK	Software library for Linear Algebra
MB	Management Board (highest decision making body of the project)
MB	Mega (= 2^{20} ~ 10^6) Bytes (= 8 bits), also MByte
MB/s	Mega (= 10^6) Bytes (= 8 bits) per second, also MByte/s
MFlop/s	Mega (= 10^6) Floating point operations (usually in 64-bit, i.e. DP) per second, also MF/s
MOOC	Massively open online Course
MoU	Memorandum of Understanding.
MPI	Message Passing Interface
NDA	Non-Disclosure Agreement. Typically signed between vendors and customers working together on products prior to their general availability or announcement.
PA	Preparatory Access (to PRACE resources)
PATC	PRACE Advanced Training Centres
PI	Principal Investigator

PRACE	Partnership for Advanced Computing in Europe; Project Acronym
PRACE 2	The upcoming next phase of the PRACE Research Infrastructure following the initial five year period.
PRIDE	Project Information and Dissemination Event
RI	Research Infrastructure
TB	Technical Board (group of Work Package leaders)
TB	Tera (= 2^{40} ~ 10^{12}) Bytes (= 8 bits), also TByte
TCO	Total Cost of Ownership. Includes recurring costs (e.g. personnel, power, cooling, maintenance) in addition to the purchase cost.
TDP	Thermal Design Power
TFlop/s	Tera (= 10^{12}) Floating-point operations (usually in 64-bit, i.e. DP) per second, also TF/s
Tier-0	Denotes the apex of a conceptual pyramid of HPC systems. In this context the Supercomputing Research Infrastructure would host the Tier-0 systems; national or topical HPC centres would constitute Tier-1
UNICORE	Uniform Interface to Computing Resources. Grid software for seamless access to distributed resources.

List of Project Partner Acronyms

BADW-LRZ	Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften, Germany (3rd Party to GCS)
BILKENT	Bilkent University, Turkey (3rd Party to UHEM)
BSC	Barcelona Supercomputing Center - Centro Nacional de Supercomputacion, Spain
CaSToRC	The Computation-based Science and Technology Research Center (CaSToRC), The Cyprus Institute, Cyprus
CCSAS	Computing Centre of the Slovak Academy of Sciences, Slovakia
CEA	Commissariat à l'Energie Atomique et aux Energies Alternatives, France (3rd Party to GENCI)
CENAERO	Centre de Recherche en Aéronautique ASBL, Belgium (3rd Party to UANTWERPEN)
CESGA	Fundacion Publica Gallega Centro Tecnológico de Supercomputación de Galicia, Spain, (3rd Party to BSC)
CINECA	CINECA Consorzio Interuniversitario, Italy
CINES	Centre Informatique National de l'Enseignement Supérieur, France (3 rd Party to GENCI)
CNRS	Centre National de la Recherche Scientifique, France (3 rd Party to GENCI)
CSC	CSC Scientific Computing Ltd., Finland
CSIC	Spanish Council for Scientific Research (3rd Party to BSC)
CYFRONET	Academic Computing Centre CYFRONET AGH, Poland (3rd Party to PNSC)
DTU	Technical University of Denmark (3rd Party of UCPH)
EPCC	EPCC at The University of Edinburgh, UK
EUDAT	EUDAT OY
ETH Zurich (CSCS)	Eidgenössische Technische Hochschule Zürich – CSCS, Switzerland
GCS	Gauss Centre for Supercomputing e.V., Germany
GÉANT	GÉANT Vereniging
GENCI	Grand Equipement National de Calcul Intensif, France
GRNET	National Infrastructures for Research and Technology, Greece

ICREA	Catalan Institution for Research and Advanced Studies (3rd Party to BSC)
INRIA	Institut National de Recherche en Informatique et Automatique, France (3rd Party to GENCI)
IST-ID	Instituto Superior Técnico for Research and Development, Portugal (3rd Party to UC-LCA)
IT4I	Vysoka Skola Banska - Technicka Univerzita Ostrava, Czech Republic
IUCC	Machba - Inter University Computation Centre, Israel
JUELICH	Forschungszentrum Jülich GmbH, Germany
KIFÜ (NIIFI)	Governmental Information Technology Development Agency, Hungary
KTH	Royal Institute of Technology, Sweden (3rd Party to SNIC-UU)
KULEUVEN	Katholieke Universiteit Leuven, Belgium (3rd Party to UANTWERPEN)
LiU	Linkoping University, Sweden (3rd Party to SNIC-UU)
MPCDF	Max Planck Gesellschaft zur Förderung der Wissenschaften e.V., Germany (3rd Party to GCS)
NCSA	NATIONAL CENTRE FOR SUPERCOMPUTING APPLICATIONS, Bulgaria
NTNU	The Norwegian University of Science and Technology, Norway (3rd Party to SIGMA2)
NUI-Galway	National University of Ireland Galway, Ireland
PRACE	Partnership for Advanced Computing in Europe aisbl, Belgium
PSNC	Poznan Supercomputing and Networking Center, Poland
SDU	University of Southern Denmark (3rd Party to UCPH)
SIGMA2	UNINETT Sigma2 AS, Norway
SNIC-UU	Uppsala Universitet, Sweden
STFC	Science and Technology Facilities Council, UK (3rd Party to UEDIN)
SURF	SURF is the collaborative organisation for ICT in Dutch education and research
TASK	Politechnika Gdańska (3rd Party to PNSC)
TU Wien	Technische Universität Wien, Austria
UANTWERPEN	Universiteit Antwerpen, Belgium
UC-LCA	Universidade de Coimbra, Labotatório de Computação Avançada, Portugal
UCPH	Københavns Universitet, Denmark
UEDIN	The University of Edinburgh
UHEM	Istanbul Technical University, Ayazaga Campus, Turkey
UIBK	Universität Innsbruck, Austria (3rd Party to TU Wien)
UiO	University of Oslo, Norway (3rd Party to SIGMA2)
UL	UNIVERZA V LJUBLJANI, Slovenia
ULIEGE	Université de Liège; Belgium (3rd Party to UANTWERPEN)
U Luxembourg	University of Luxembourg
UM	Universidade do Minho, Portugal, (3rd Party to UC-LCA)
UmU	Umea University, Sweden (3rd Party to SNIC-UU)
UnivEvora	Universidade de Évora, Portugal (3rd Party to UC-LCA)
UnivPorto	Universidade do Porto, Portugal (3rd Party to UC-LCA)
UPC	Universitat Politècnica de Catalunya, Spain (3rd Party to BSC)
USTUTT-HLRS	Universitaet Stuttgart – HLRS, Germany (3rd Party to GCS)
WCSS	Politechnika Wroclawska, Poland (3rd Party to PNSC)

Executive Summary

Applications Enabling and Porting Services in Work Package 7 (WP7) of PRACE-6IP provides HPC enabling support for the applications of European researchers and small and medium enterprises to ensure the applications can effectively exploit the various PRACE HPC systems. Most of the activities are ongoing and were already established and described in deliverables of former PRACE-IP projects. There are four activities described in this deliverable:

T7.1 Applications Enabling Services for Preparatory Access:

This activity provides code enabling and optimisation to European researchers as well as industrial projects to make their applications ready for Tier-0 systems. Projects can continuously apply for such services via the Preparatory Access Calls Type C (PA Type C) and Type D (PA Type D) with a cut-off every three months for evaluation of the proposals. PA Type C provides support and access to a PRACE Tier-0 system while PA Type D provides support and access to a PRACE Tier-1 system to finally reach Tier-0 scalability. In total ten Preparatory Access cut-offs have been carried out in PRACE-6IP and 21 projects received support within the context of the project.

Beside the statistical overview about the cut-offs and all supported PRACE PA Type C and Type D projects, the report focuses on the optimisation work and results gained by the completed projects in PRACE-6IP. In total twelve PA projects have finished their work since the last deliverable D7.1 of PRACE-6IP [4] and are reported within this deliverable.

T7.2 Applications Enabling Services for Industry (SHAPE):

This activity has continued the support for SHAPE (the SME HPC Adoption Programme in Europe) throughout the PRACE-6IP project. SHAPE aims to assist European SMEs by offering them a free, low risk opportunity to try applying HPC to their business processes with an aim to increase their competitiveness. Calls are issued twice per year with successful applicants receiving support effort from a PRACE HPC expert and access to machine time at a PRACE centre.

Throughout PRACE-6IP SHAPE has focussed on increasing the number of proposals, as well as increasing the number of countries benefiting from SHAPE support. This has been a great success. Since regular calls were introduced and up until the SHAPE 10 call, the average number of proposals received per call was just under 6. From SHAPE 11 onwards proposal numbers have increased significantly with the average number of proposals between SHAPE 11 and SHAPE 14 having more than doubled to just over 12. The SHAPE 12 call received a record 16 proposals, then beaten by the SHAPE 14 call receiving a record 19 proposals. This has continued to be greatly assisted by the employment in September 2019 of a PRACE Industry Liaison Officer who has engaged with hundreds of SMEs via face-to-face meetings and industry-specific trade shows, as well as advising on publicity and communication with SMEs.

Up to the SHAPE 9 call SHAPE had received proposals from 17 of the 26 eligible countries. The SHAPE+ initiative was then introduced where an allocation of effort is provided to support SMEs from countries new to SHAPE in cases where the local partners do not have existing SHAPE effort. This produced a noticeable increase in the number of countries participating with 23 of the eligible countries now having produced proposals.

The success of SHAPE can also be demonstrated in more qualitative ways. A set of Success Stories have been produced and displayed on the PRACE website, and successful past projects have been highlighted to potential applicants via flyers created from projects from previous calls. In addition, we received an overwhelmingly positive response when surveying SMEs a year or so after their project has ended. The results of this can be seen within this deliverable.

T7.3 DECI Management and Applications Porting:

The DECI (Distributed European Computing Initiative) programme provides access to Tier-1 resources across Europe via a resource exchange programme and for the last few years calls have been issued yearly. Support for DECI has continued throughout PRACE-6IP.

Calls remain popular with researchers and since the programme began, over 1200 proposals have been received with the most recent two calls receiving 68 and 81 proposals respectively from 25 different countries resulting in 48 and 55 projects respectively. The most recent call awarded 250 M machine core hours to projects from 17 different countries on systems from 10 different countries.

T7.5 Enhancing the High-Level Support Teams:

T7.5 is a new task implemented in PRACE-6IP. It provides additional effort to enhance the work of the PRACE High-Level Support Teams (HLSTs). Under PRACE 2, there are coordinated HLSTs at each Tier-0 centre that provide users with support for code enabling and scaling out of scientific applications and methods, as well as for Research & Development on code refactoring on the Tier-0 systems. The HLSTs will enhance the scientific output of the Tier-0 systems through the provision of level 3 (midterm activities) support activities. Task 7.5 supplements those activities and extends this work with specific expertise from the other PRACE centres. This ensures sharing of expertise across PRACE to maximise the benefits to users of the PRACE systems. Task 7.5 will work with the HLSTs to extend and enhance their activities supporting Tier-0 users, and also Tier-1 intensive users targeting Tier-0, in order to maximise the scientific output of the Tier-0 systems.

During the first half of the project, we focused on defining the process of project selection, based on criteria that meet what Tier-0 systems can provide to the users (scalability, new architectures, memory/storage …) and a few projects started.

During the second half of the project the activity continued on petascale and pre-exascale systems, which were the main targets for the activity, and the effort of the partners involved in the task help users to leverage their application to larger systems, enabling them to perform larger simulation in the most efficient manner, as recommended by the PRACE experts that provided support during the activity.

1 Introduction

PRACE offers a wide range of different Tier-0 and Tier-1 HPC architectures to the scientific community as well as to innovative projects from industry. The efficient usage of such systems places high demands on the used software packages and in many cases advanced optimisation work has to be applied to the code to make efficient use of the provided supercomputers. The complexity of supercomputers requires a high level of experience and advanced knowledge of different concepts regarding programming techniques, parallelisation strategies, etc. Such demands often cannot be met by the applicants themselves and thus special assistance by supercomputing experts is essential.

PRACE offers such a service through the Preparatory Access Calls. PA Type C and PA Type D are managed by Task 7.1. This includes the evaluation of the PA proposals as well as the assignment of PRACE experts to these proposals. Furthermore, the support itself is provided and monitored within this task. Section 2 gives a more detailed description of PA and some facts on the usage of PA Type C and PA Type D in PRACE-6IP are listed in Section 2.1. The review process, the assignment of PRACE experts to the projects and the monitoring of the support work are detailed in Section 2.2, Section 2.3, and Section 2.4 respectively. The contents of Sections 2.2 - 2.4 can already be found in deliverable D7.1 of PRACE-6IP [4]. They are repeated here for completeness and the benefit of the reader. Section 2.5 gives an overview about the Preparatory Access projects covered by this report in the frame of PRACE-6IP. The dissemination of the PA call is described in Section 2.6. Finally, the work done within the projects along with the outcome of the optimisation work is presented in Sections 2.7 - 2.13.

The next part of this deliverable reports on SHAPE (SME HPC Adoption Programme in Europe). SHAPE is a pan-European programme set up to support the adoption of High Performance Computing (HPC) by European small to medium-size enterprises (SMEs). It was introduced by PRACE as a pilot call in 2013 issued under the PRACE-3IP European Union funded project and has continued through to the present PRACE-6IP project.

SHAPE was originally presented in the PRACE-3IP Deliverable [24]. SHAPE aims to raise awareness of HPC within European SMEs providing them with the necessary expertise and resources to assess the innovation possibilities opened up by HPC, with the aim of increasing SMEs' competitiveness. SHAPE has now run 14 calls, the most recent of which received a record 19 proposals from 11 different countries.

In Section 3.1 we give an overview of the SHAPE programme followed by a report on the SHAPE+ initiative in Section 3.2. The present status of SHAPE is described in Section 3.3. In Section 3.4 we report on the very successful survey of SMEs and partners giving an opportunity to see what the SMEs get from a SHAPE project. Section 3.5 then describes the more recently finished or on-going projects. Finally we report on the most recent call (SHAPE 14) and a look towards the EuroHPC era in Section 3.6.

The next part of the deliverable is the report on the DECI (Distributed European Computing Initiative) resource exchange programme. In Section 4.1 we give an overview of the programme and the present status including some statistics and enabling work carried out from the most recent calls is given in Section 4.2. In Section 4.3 we talk about DECI as we move through the PRACE-6IP extension towards the EuroHPC era.

In the final part T7.5 is presented, which is a new task implemented in PRACE-6IP. It provides additional effort to enhance the work of the PRACE High-Level Support Teams (HLSTs). Under PRACE 2, there are coordinated HLSTs at each Tier-0 centre that provide users with

support for code enabling and scaling out of scientific applications and methods, as well as for Research & Development on code refactoring on the Tier-0 systems. The HLSTs will enhance the scientific output of the Tier-0 systems through the provision of level 3 (midterm activities) support activities. Task 7.5 supplements those activities and extends this work with specific expertise from the other PRACE centres. This will ensure sharing of expertise across PRACE to maximise the benefits to users of the PRACE systems. Task 7.5 will work with the HLSTs to extend and enhance their activities supporting Tier-0 users, and also Tier-1 intensive users targeting Tier-0, in order to maximise the scientific output of the Tier-0 systems.

2 T7.1 Applications Enabling Services for Preparatory Access

Access to PRACE Tier-0 systems is managed through PRACE regular calls, which are issued twice a year [2]. To apply for Tier-0 resources the application must meet technical criteria concerning scaling capability, memory requirements, and runtime set up. There are many important scientific and commercial applications that do not meet these criteria today. To support researchers PRACE offers the opportunity to test and optimise their applications on the envisaged Tier-0 system prior to applying for a regular production project. This is the purpose of the Preparatory Access (PA) Call. The PA Call allows for submission of proposals at any time. Depending on the PA scheme, the review of these proposals takes place directly after the submission of the proposal (Type A and B) or at a fixed date every three months (Type C and D). This procedure is also referred to as cut-off [3]. It is possible to choose between four different types of access:

- Type A is meant for code scalability tests, the outcome of which is to be included in the proposal in a future PRACE regular call. Users receive a limited number of core hours; the allocation period is two months.
- Type B is intended for code development and optimisation by the user. Users also get a small number of core hours; the allocation period is six months.
- Type C is also designed for code development and optimisation with the core hours and the allocation period being the same as for Type B. The important difference is that Type C projects receive special assistance by PRACE experts from the PRACE-IP project to support the optimisation. As well as access to the Tier-0 systems, the applicants also apply for one to six person months (PMs) of supporting work to be performed by PRACE experts.
- Type D allows PRACE users to start a code adaptation and optimisation process on a PRACE Tier-1 system. PRACE experts help in the system selection process. In addition to Tier-1 computing time, the PRACE user will also receive Tier-0 computing time towards the end of the project in the form of a PA Type A project to test the scalability improvements. The work is supported by PRACE experts similar to Type C. The maximum duration of Type D projects is twelve months.

Task 7.1 of the PRACE project covers the expert assistance in context of PA Type C and D. PA Type A and B projects are directly handled by the different hosting sites without involvement of the PRACE-IP project.

2.1 Cut-off statistics

Since the last PRACE-6IP deliverable D7.1 [22] four cut-offs for PA took place in between October 2020 and November 2021 resulting in three new projects. In total, 21 projects were supported by PRACE-6IP T7.1. Seven of them started already during PRACE-5IP or were established with the frame of a PRACE-5IP cut-off. All of those overlapping projects were taken over by PRACE-6IP. All projects, except one, could be fully finalised in the frame of PRACE-6IP. Nine were already reported in deliverable D7.1 [22]. All remaining will be reported within this deliverable.

Within the reporting period between October 2020 and November 2021, only one proposal had to be rejected, because it did not fit into the scope of the preparatory access call as no general

HPC support but a dedicated technical contact of the involved hosting site was required. To still support the project, it was moved to PA Type B and received support by the individual hosting sites instead.

A cut-off for new PA Type C or D projects in December 2021 was not planned, due to the end of the reporting period and the original PRACE-6IP runtime. PA Type A and B will continue unchanged.

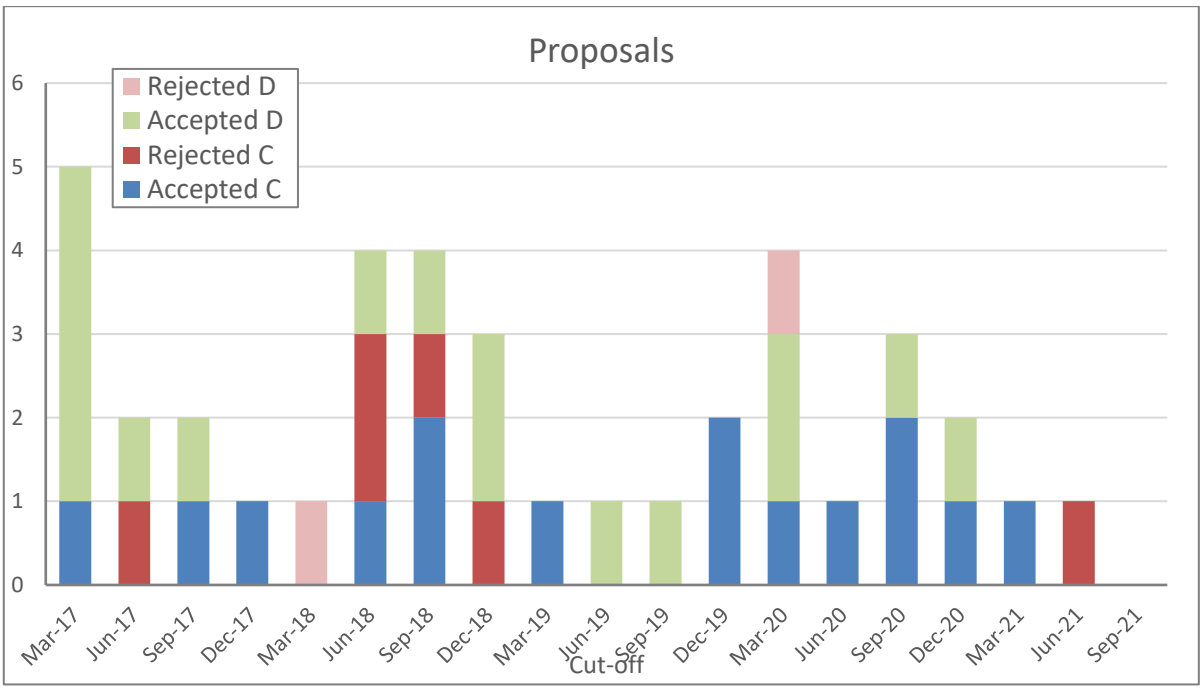


Figure 1: Number of proposals for PA type C and type D per cut-off.

Figure 1 presents the number of proposals that have been accepted or rejected, respectively for each cut-off. Within the frame of PRACE-6IP reporting period (April 2019 – December 20221) 14 out of 16 projects were accepted.

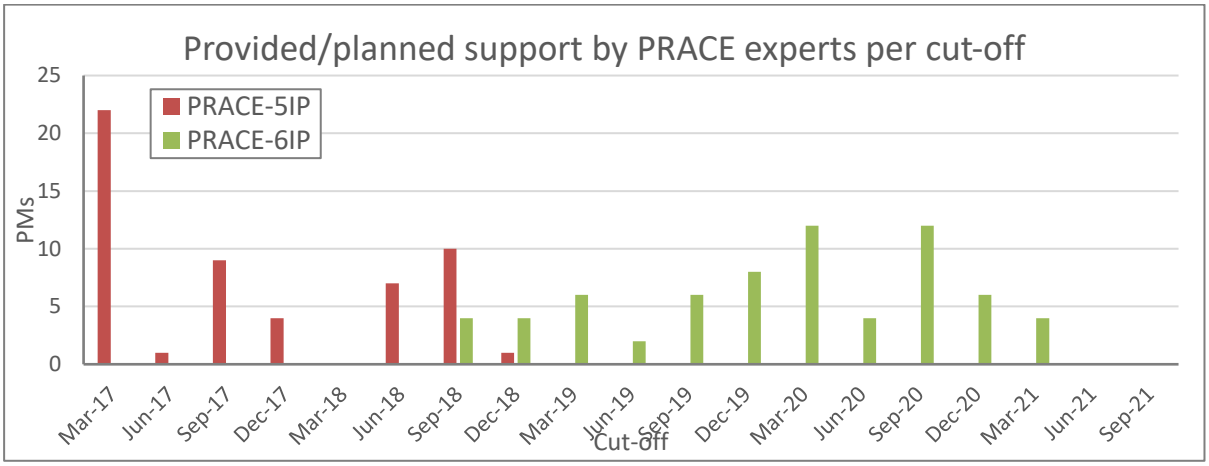


Figure 2: Amount of PMs assigned to PA projects per cut-off.

Figure 2 gives an overview of the amount of PMs from the PRACE project assigned to the projects per cut-off. In total 68 PMs were made available to these projects within the context of PRACE-6IP.

Finally, Figure 3 provides an overview of the scientific fields, which are covered by the supported projects in PRACE-6IP.

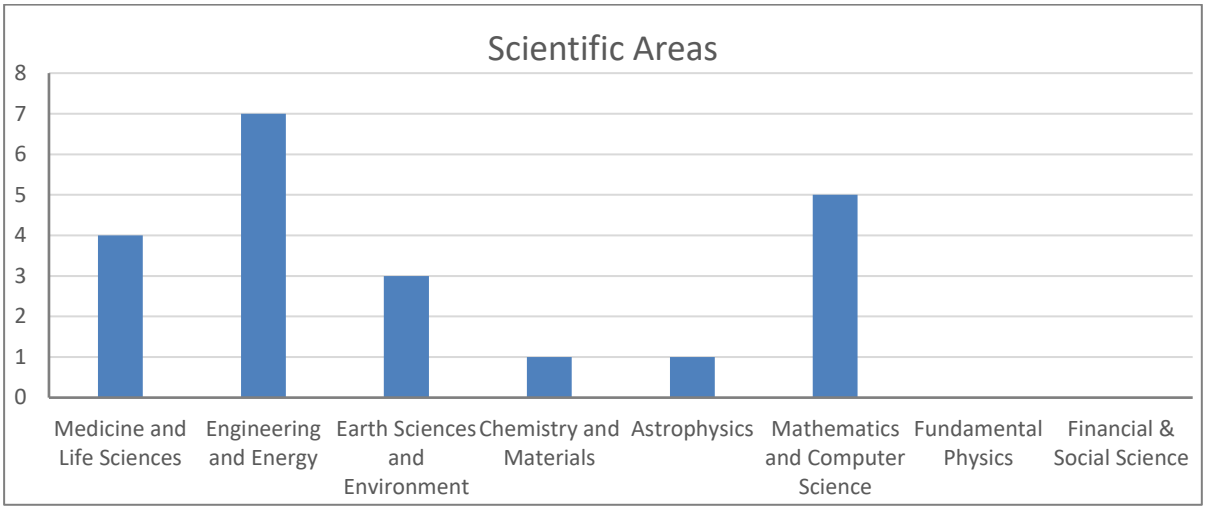


Figure 3: Number of projects per scientific field.

2.2 Review Process

The organisation of the review procedure, the assignment of PRACE collaborators and the supervision of the PA Type C and D projects are managed by task 7.1. In this section, the review process for the preparatory access proposals of Type C and Type D is explained.

All preparatory access proposals undergo a technical review performed by technical staff of the hosting sites to ensure that the underlying codes are in principle able to run on the requested system. For PA C projects, the technical review starts directly after the cut-off. For PA D projects, the technical review is done after the Tier-1 system is finally selected.

In parallel, all projects are additionally reviewed by WP7 staff in order to assess their optimisation requests. Each proposal is assigned to two WP7 reviewers. The review is performed by PRACE partners who all have a strong background in supercomputing. The task leader has the responsibility to contact them to launch the review process. As the procedure of reviewing proposals and establishing the collaboration of submitted projects and PRACE experts takes place only four times a year, it is necessary to keep the review process swift and efficient. A close collaboration between PRACE aisbl, T7.1 and the hosting sites is important in this context. The process for both the technical and the WP7 review is limited to six weeks. In close collaboration with PRACE aisbl and the hosting sites, the whole procedure from PA cut-off to project start on PRACE supercomputing systems is completed in less than six weeks as well. Due to different availabilities the PI as well as the PRACE expert can change the final start date if necessary.

Based on the proposals the Type C and Type D reviewers need to focus on the following aspects:

- Does the project require support for achieving production runs on the chosen architecture?
- Are the performance problems and their underlying reasons well understood by the applicant?
- Is the amount of support requested reasonable for the proposed goals?

- Will the code optimisation be useful to a broader community, and is it possible to integrate the achieved results during the project in the main release of the code(s)?
- Will there be restrictions in disseminating the results achieved during the project?

For Type D, the reviewer should also make suggestions for the Tier-1 selection process. The 7.1 task leader finally selects the Type D Tier-1 computing site based on the suggestions of the PI, the potential PRACE collaborator, the reviewers and the computing site availability.

In addition to the WP7 reviews, the task leader evaluates whether the level and type of support requested is still available within PRACE. Finally, the recommendation from WP7 to accept or reject the proposal is made.

The outcome is communicated to the applicant through PRACE aisbl. Here also the PRACE Board of Directors is informed. Approved proposals receive the contact data of the assigned PRACE collaborators. Rejected projects are provided with further advice on how to address the shortcomings.

2.3 Assigning of PRACE collaborators

To ensure the success of the projects it is essential to assign suitable experts from the PRACE project. Based on the described optimisation issues and support requests from the proposal experts are thus chosen who are most familiar with the subject matter.

This is done in two steps: first, summaries of the proposals describing the main optimisation issues are distributed via corresponding mailing lists. Here, personal data is explicitly removed from the reports to maintain the anonymity of the applicants. Interested experts can get in touch with the task leader offering to work on one or more projects.

There is one exception to the procedure when a proposal has a close connection to a PRACE site, which has already worked on the code or was in close contact with the involved applicant. In this case this site is asked first if they are able to extend the collaboration in the context of the new PA Type C or PA Type D project.

Should the response not be sufficient to cover the support requirements of the projects, the task leader contacts the experts directly and asks them to contribute.

The assignment of PRACE experts takes place concurrently to the review process so that the entire review can be completed within six weeks. This has proven itself to be a suitable approach, as the resulting overhead is negligible.

As soon as the review process is finished, the support experts are introduced to the PIs and can start the work on the projects. The role of the PRACE collaborator includes the following tasks:

- Formulating a detailed work plan together with the applicant;
- Participating in the optimisation work;
- Reporting the status back to the task leader of T7.1;
- Participating in the writing of the final report together with the PI (the PI has the main responsibility for this report), due at project end and requested by the PRACE office;
- Writing a final expert report, due at project end and requested by the T7.1 task leader to be used in the PRACE-6IP deliverable work;
- Writing an optional white paper containing the results, which is published on the PRACE web site [1].

2.4 Monitoring of projects

Task 7.1 includes the supervision of the Type C and Type D projects. This is challenging as the projects' durations (six months for PA Type C and twelve months for PA Type D) and the intervals of the cut-offs (three months) are not synchronised. Due to this, projects do not necessarily start and end at the same time but overlap, i.e. at each point in time different projects might be in different phases. To solve this problem the T7.1 task leader gives a status overview in a monthly WP7 conference call to address all PRACE collaborators who are involved in these projects. In addition the task leader monitors all relevant project deadlines to inform the PRACE experts concerning reporting periods.

The T7.1 task leader is also available to address urgent problems and additional phone conferences are held in such cases.

2.5 PRACE Preparatory Access projects covered by this report

The support for Preparatory Access projects has been and is part of all PRACE projects (PRACE-1IP, -2IP, -3IP, -4IP, -5IP, -6IP). Therefore, finalised projects will be reported continuously for a given reporting period.

The timeline of all projects supported by PRACE-6IP is shown in the chart of Figure 4. The chart shows the time span of each project. Projects supported by PRACE-5IP are shown in red. The projects which received support by PRACE-6IP are shown in green. The slightly different starting dates of the projects per cut-off are the result from the decisions made by the hosting members, the PI and the PRACE expert, which determine the exact start of each individual project. Changes in the project runtime were made on individual request by the applicant or the PRACE expert when necessary.

Projects completed before September 2020 were already reported in previous deliverables. This report will cover eleven projects which were finalised in between September 2020 and December 2021. One remaining ongoing project will be covered by an intermediate status report. Table 1 provides an overview about the projects reported in this deliverable.

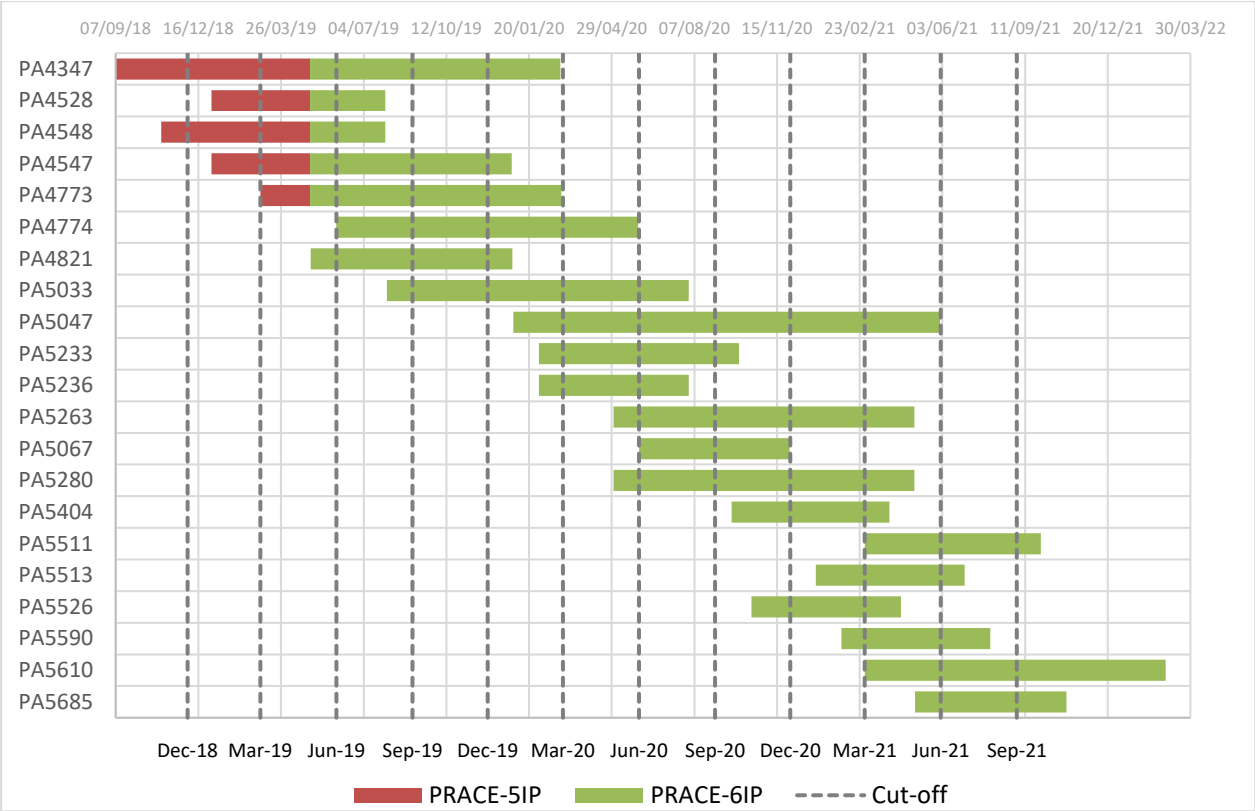


Figure 4: Timeline of the PA projects.

Cut-off September 2019	
Title	**Next steps for scalable Delft3D FM for efficient modelling of shallow water and transport processes**
Type	D
Project leader	Menno Genseberger
PRACE expert	Andrew Emerson, Maxime Mogé
PRACE facility	CARTESIUS, MARCONI
PA number	2010PA5047
Project's start	01-January-2020
Project's end	31-May-2021

Cut-off December 2019	
Title	**Load Balancing of Molecular Properties Calculations In VeloxChem Program**
Type	C
Project leader	Zilvinas Rinkevicius
PRACE expert	Isabelle Dupays
PRACE facility	SUPERMUC-NG
PA number	2010PA5233

Cut-off December 2019	
Project's start	01-February-2020
Project's end	30-September-2020

Cut-off March 2020	
Title	**Enhancing Parallelism in MAGMA2**
Type	D
Project leader	Stephan Rosswog
PRACE expert	Jing Gong
PRACE facility	BESCOW
PA number	2010PA5263
Project's start	01-May-2020
Project's end	30-April-2021
Title	**XDEM4HPC: eXtended Discrete Element Method for High-Performance Computing**
Type	C
Project leader	Alban Rousset
PRACE expert	Pedro Ojeda May
PRACE facility	MARENOSTRUM
PA number	2010PA5067
Project's start	01-June-2020
Project's end	30-November-2020
Title	**Scalability of Automatic Design Optimisation Algorithms**
Type	D
Project leader	Zilvinas Rinkevicius
PRACE expert	Joris Coddé, Engelbert Tijskens
PRACE facility	BRENIAC
PA number	2010PA5280
Project's start	01-May-2020
Project's end	30-April-2021

Cut-off June 2020	
Title	**HOVE3 Higher-Order finite-Volume unstructured code Enhancement for compressible turbulent flows**
Type	C

Cut-off June 2020	
Project leader	Panagiotis Tsoutsanis
PRACE expert	Anastasiia Shamakina, Anna Mack
PRACE facility	HAWK
PA number	2010PA5404
Project's start	21-September-2020
Project's end	31-March-2021

Cut-off September 2020	
Title	**Use of HPC for optimization of drinking water distribution networks**
Type	D
Project leader	Mario Castro-Gama
PRACE expert	Jean-Christophe Pénalva
PRACE facility	OCCIGEN
PA number	2010PA5511
Project's start	01-March-2021
Project's end	30-September-2021
Title	**FESOM2 Finite volumE Sea ice Ocean Model enhancement**
Type	C
Project leader	Sergey Danilov
PRACE expert	Mario Acosta
PRACE facility	MARENOSTRUM
PA number	2010PA5513
Project's start	01-January-2021
Project's end	30-June-2021
Title	**Fast MDS**
Type	C
Project leader	Alain Franc
PRACE expert	Olivier Coulaud
PRACE facility	HAWK
PA number	2010PA5526
Project's start	15-October-2020
Project's end	14-April-2021

Cut-off Decmber 2020	
Title	**Parallel high order finite element solver for the simulation of nanoscale light-matter interaction in disordered media**
Type	C
Project leader	Gian Luca Lippi
PRACE expert	Stéphane Lanteri
PRACE facility	MARCONI100
PA number	2010PA5590
Project's start	01-February-2021
Project's end	31-July-2021
Title	**PRACE QBee - Towards parallel quantum circuits for now**
Type	D
Project leader	Koen Bertels
PRACE expert	João Cardoso
PRACE facility	GALILEO, JUWELS Cluster
PA number	2010PA5610
Project's start	01-March-2021
Project's end	28-February-2022 (ongoing)

Cut-off March 2021	
Title	**Towards prototypical Rayleigh numbers in molten pool convection**
Type	C
Project leader	Walter Villanueva
PRACE expert	Jing Gong
PRACE facility	JUWELS Booster
PA number	2010PA5685
Project's start	01-May-2021
Project's end	31-October-2021

Table 1: PA Projects, which are reported in this deliverable.

2.6 Dissemination

New PA Cut-offs are normally announced on the PRACE website [1] and through the official PRACE Twitter channel.

Sites are also asked to distribute an email to their users to advertise preparatory access and especially the possibility of the dedicated support.

Each successfully completed project should be made known to the public and therefore the PRACE collaborators are asked to write a white paper about the optimisation work carried out. These white papers are published on the PRACE web page [5] and are also referenced by this deliverable.

2.7 Cut-off September 2019

This section and the following sub-sections describe the optimisations performed on the preparatory access projects. The projects are listed in accordance with the cut-off dates in which they applied. General information regarding the optimisation work done as well as the achieved results are presented here.

2.7.1 Next steps for scalable Delft3D FM for efficient modelling of shallow water and transport processes, 2010PA5047

Overview:
Forecasting of flooding, morphology, and water quality in coastal and estuarine areas, rivers, and lakes is of great importance for society. To tackle this, the modelling suite Delft3D, was developed by Deltares (independent non-profit institute for applied research in the field of water and subsurface). Delft3D is used worldwide. Delft3D has been open-source since 2011. It consists of modules for modelling hydrodynamics, waves, morphology, water quality, and ecology.

Currently, for Delft3D there is a transition from the shallow water solver Delft3D-FLOW for structured computational meshes to D-Flow FM (Flexible Mesh) for unstructured computational meshes. D-Flow FM will be the main computational core of the Delft3D Flexible Mesh Suite [6][7]. For typical real-life applications there is urgency to make D-Flow FM also more efficient and scalable for high performance computing. The aim of the project was to make significant progress towards Tier-0 systems for the shallow water and transport solvers in the Delft3D FM Suite.

This project was a continuation of a previous preparatory access type D project carried out by SURF, CINECA, and Deltares. That previous project yielded new insights and it also gave the opportunity to test possible improvements in D-Flow FM for real-life applications. Earlier testing did not go beyond a few hundred MPI processes, now we successfully ran representative simulations on up to several thousands of MPI processes. For the selected test cases from the previous preparatory access type D project, it has become clear which further steps have to be taken to be able to run the software efficiently on the Tier-0 systems. That was the subject of this new preparatory access type D project. It focused on three-dimensional model applications (instead of two-dimensional, depth averaged ones) and water quality applications (and not only hydrodynamics as in the previous project).

Scalability results:
The scalability testing focuses on the time loop, excluding the initialisation and finalisation phases of the simulation.

Additionally, for each test case the performances and scaling of the main components of the time loop have been measured separately. The main components are the following:

- transport: subroutine `transport`
- furu: subroutine `furu`
- solve: subroutine `solve_matrix` - linear solver for the implicit solver
- processes: subroutine `fm_wq_processes_step`
- inistep: subroutine `flow_initimestep`

Different test cases were evaluated:

- The IJSM3D and NorthSea_3D_WE test cases show a decent scaling up to 16 nodes / 512 cores, with an efficiency of 0.53 for both cases.
- The RMM test case shows a decent scaling only up to 8 nodes / 256 cores, with an efficiency of 0.53.
- The VZM test case shows a decent scaling only up to 4 nodes / 128 cores, with an efficiency of 0.54.
- The VZM test case is too small to scale to large core counts and was left out of the further performance analysis and optimisation.

In Figure 5 and Figure 6, the IJSM3D and RMM test case before optimisation are shown. Additional MPI analyses were performed on these two test cases using the "MPI analysis" from Intel APS [18].
Based on the preliminary conclusions for the IJSM3D model, that most time was spent in `mpi_barriers` and in `mpi_allreduce`, the corresponding routines were changed. This concerns the `update_ghost_loc` and `s1ini` routines, and the `setdt` routine for determining the time step size (see also accomplished work). In this way, three routines have been changed to optimise the MPI communications, and the performance measurements. Below in Figure 7 to Figure 10 show the MPI profiling of the IJSM3D test case with different variants of the Delft3D FM code:

- original version of the code
- version 1 with changes in routines `update_ghost_loc`
- version 2 with changes in routines `update_ghost_loc` and `s1ini`
- version 3 with changes in routines `update_ghost_loc` and `s1ini` and fixed time step

Although there are significant improvements in the MPI scaling for some components in all three optimised versions, there is no significant change in the total run time of the time loop or in the overall scaling of the time loop for the IJSM3D test cases.

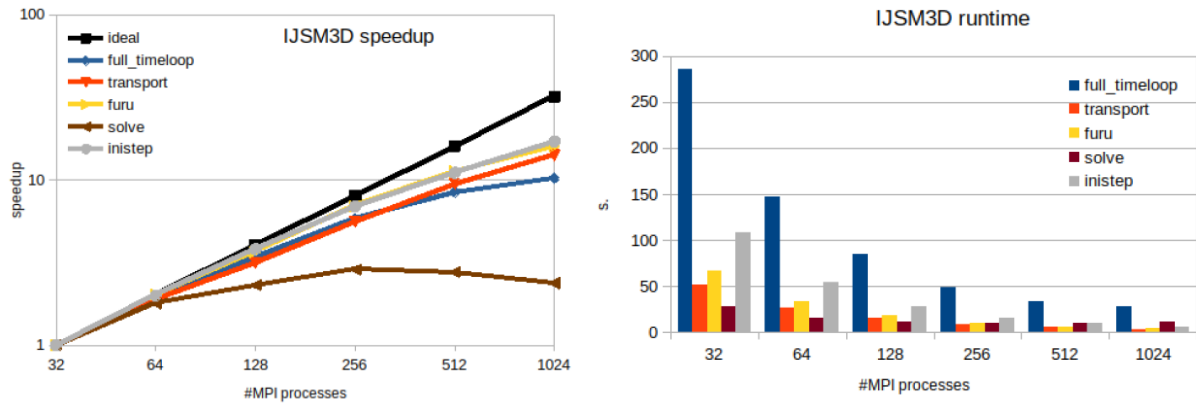


Figure 5: Speedup and runtime of full-time loop and main components of time loop for IJSM3D test case before optimisation.

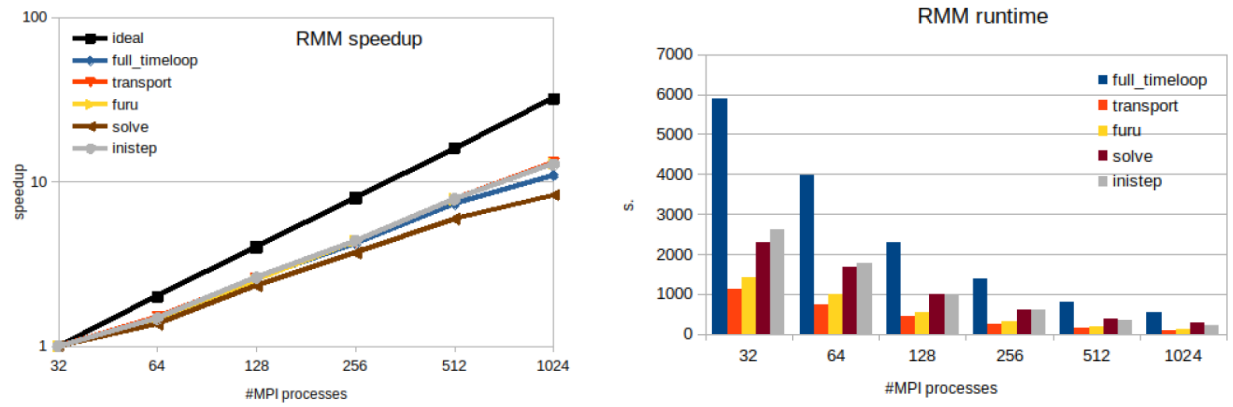


Figure 6: Speedup and runtime of full-time loop and main components of time loop for RMM test case before optimisation.

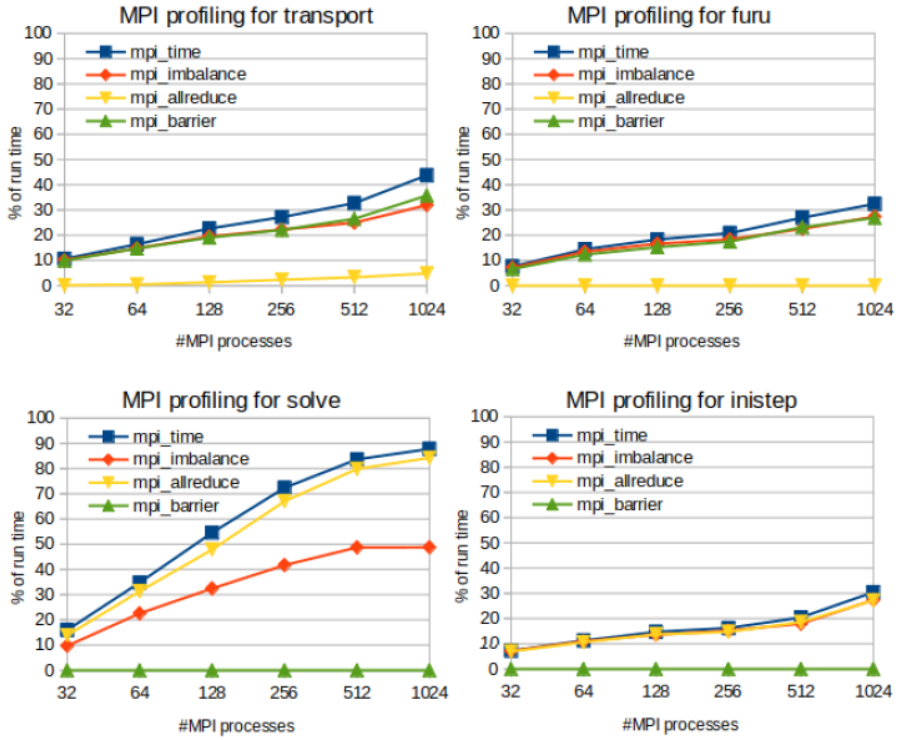


Figure 7: MPI profiling for IJSM3D test case with original version of the code

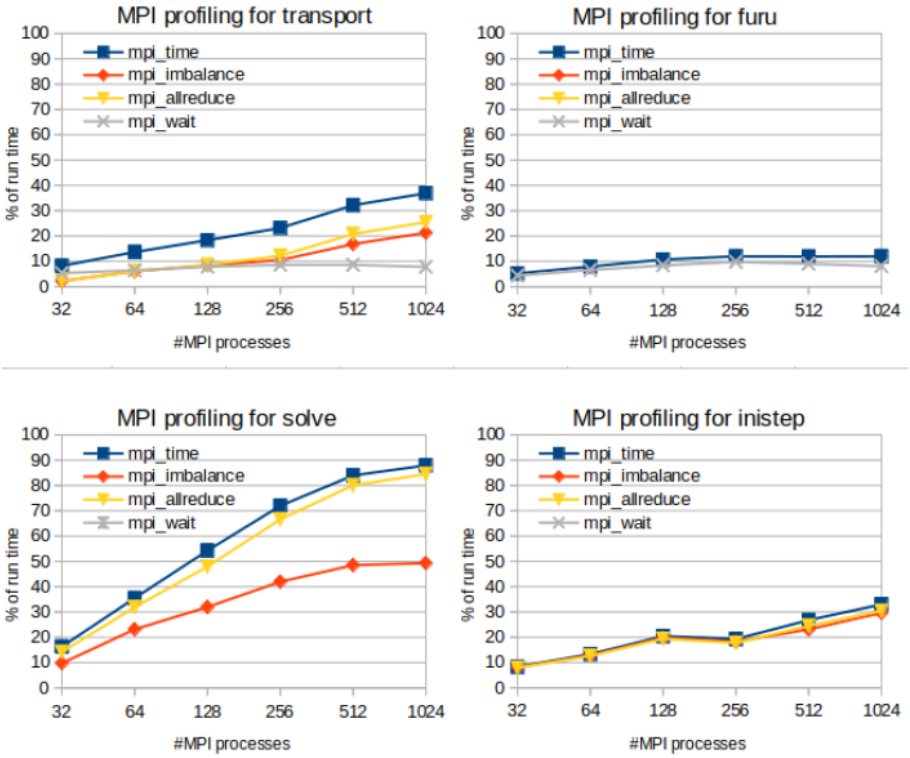


Figure 8: MPI profiling for IJSM3D test case with version 1 update_ghost_loc

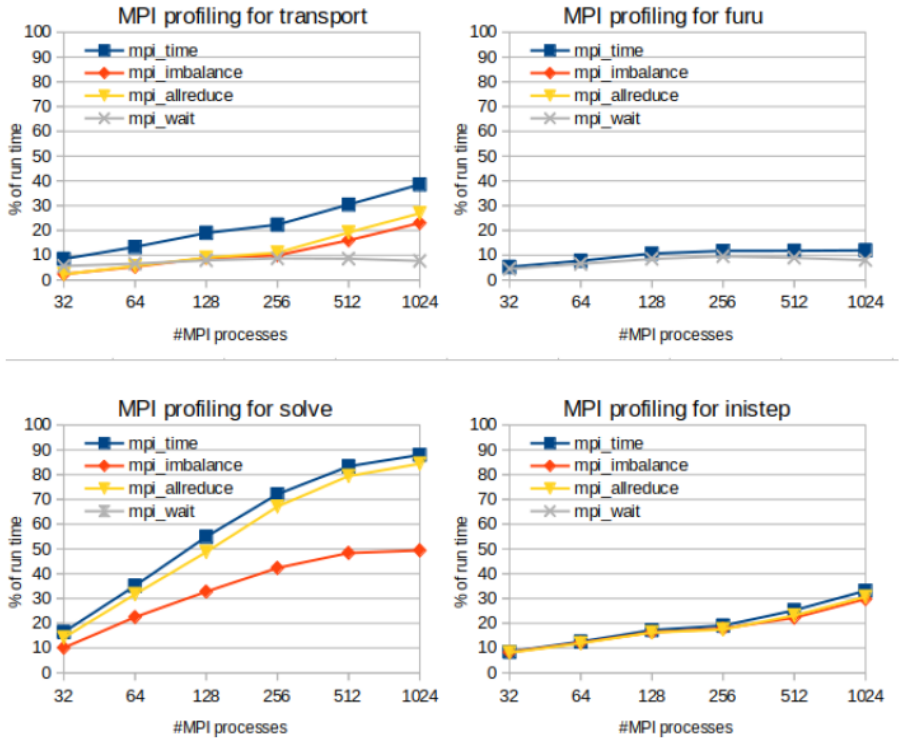


Figure 9: MPI profiling for IJSM3D test case with version 2 s1ini

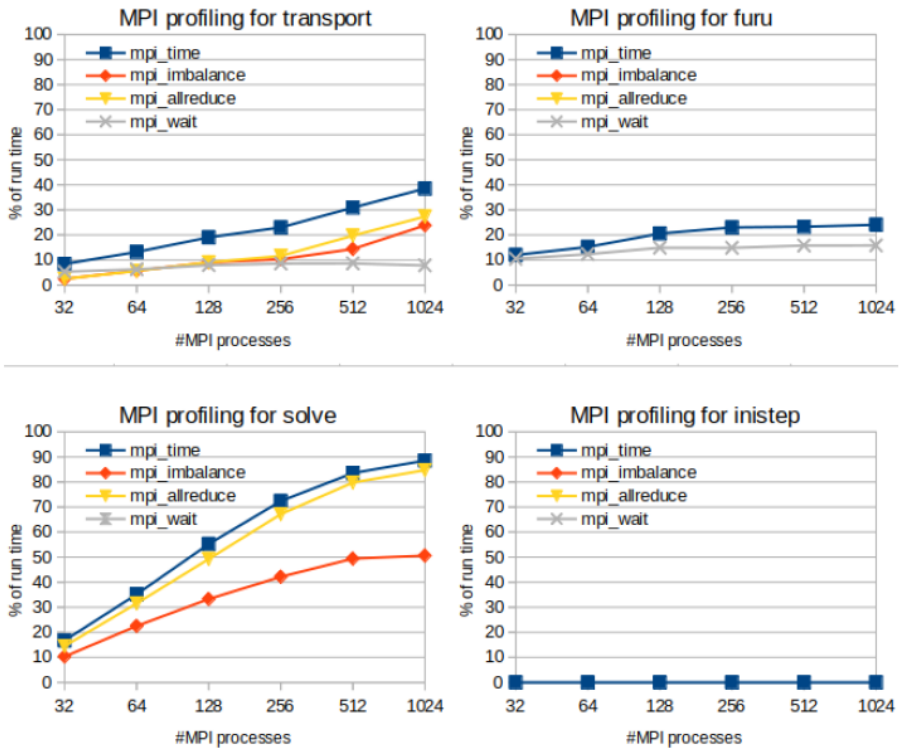


Figure 10: MPI profiling for IJSM3D test case with version 3 fixed time step

Accomplished work:

An MPI profiling together with the trace analysis showed that all test cases are MPI bound when scaling up, and that there are global synchronisations and blocking MPI calls within the time loop that account for most of the MPI time.

The MPI calls made within the time loop for all test cases have been identified, and the following optimisations have been done:

- In routine `update_ghost_loc`, an `mpi_barrier` was removed and non-blocking receive were used instead of blocking receive. The resulting version of the code is referred to as "version 1 - update_ghost_loc".
- In routine `setsorsin`, the loops and function calls were reordered to allow splitting the call to `reduce_srsn` in multiple parts in order to replace the blocking `allreduce` with a non-blocking `iallreduce` and overlap the communications with computations. The resulting version of the code, with the optimisation in bullet 1 and 2, is referred to as "version 2 - s1ini".
- For test case IJSM3D only, we tested changing the adaptive time stepping method - which is used in Delft3D FM in general - to a fixed time stepping method. Note that in general this results in a change in the numerical approach and therefore changes the results. However, for IJSM3D already a fixed time step was forced via the model settings. This was meant as a test to check the impact on the MPI communications of the local time stepping. Using a fixed time step removes one `allreduce` operation. The resulting version of the code, with the optimisation in bullet 1, 2 and 3, is referred to as "version 3 – fixed_time_step".

Unfortunately the performance of the optimised versions shown no significant improvement in the scaling or run time.

In addition it was already known from a previous project (PA3775) that the domain decomposition in Delft3D FM causes a significant imbalance in the halo regions. Since the halo region are large (depth 4), the number of halo cells becomes important relative to the number of inner cells when the number of subdomains increases, and thus the imbalance on the halo regions causes a large imbalance on the local computation domains.

A hypergraph was used to represent the computation domain instead of a graph in the original version, and the halo-aware repartitioning algorithm proposed in [19] was used.

Main results:
The latest stable version of the Delft3D FM software suite was installed on the Dutch Tier-1 supercomputer Cartesius, and the installation was tested for correctness and the results of representative tests validated by experts from Deltares. This installation is now available to all users of Cartesius, among which is a growing number of researchers using Delft3D FM. The profiling and scalability analysis of a representative set of test cases has already been used for guiding the users into making efficient use of the computing resources with a good time to solution.

The MPI optimisation showed benefit on some relevant metrics (time spent in MPI, MPI imbalance) but showed no significant benefit on the actual run time.

It is clear from traces that some synchronisations have disappeared in the optimised versions. However, using a non-blocking allreduce in `s1ini` had no visible effect, possibly because the communications are not effectively overlapped with the computation without using a communication thread, or because the amount of computation in the `s1ini` routine is insufficient for it to be visible.

With the optimised versions the imbalance between processes is even more visible than with the original versions. This imbalance is probably caused by an imbalance in the compute workload.

The attempts at improving the computational imbalance were unsuccessful. An implementation of the algorithm is not converging for an unknown reason, and the project could not get help from the original authors as they have all moved out of academia.

The domain decomposition obtained with the implementation of a halo-aware hypergraph repartitioning algorithm showed an imbalance similar to the imbalance in the graph obtained with the original domain decomposition with Metis.

2.8 Cut-off December 2019

2.8.1 Load Balancing of Molecular Properties Calculations In VeloxChem Program, 2010PA5233

Overview:
The purpose of the project was to optimise the performance of the code VeloxChem to keep a good scalability up to 100k cores. Originally, the scalability was good up to 18k cores. By having a good scalability over more cores, the application can have access to more resources (like memory) hence the size of the physical systems studied and/or the precision of the methods can be increased.

The code is parallelised with MPI and OpenMP. The strategy is to have one MPI task per node and OpenMP threads on one node. One of the possible bottlenecks identified by the developers is the MPI communications between nodes.

The lockdown for several months in France made it difficult to work with good conditions for some of the people involved in the project. This is one of the reasons the project was not able to proceed with the subsequent parts of the project except the profiling work.

Scalability results:

Scaling results on two test cases were performed: The first test case is for helicene (large system) and shown in Figure 11 while the second test case is for fullerene (medium system), shown in Figure 12.

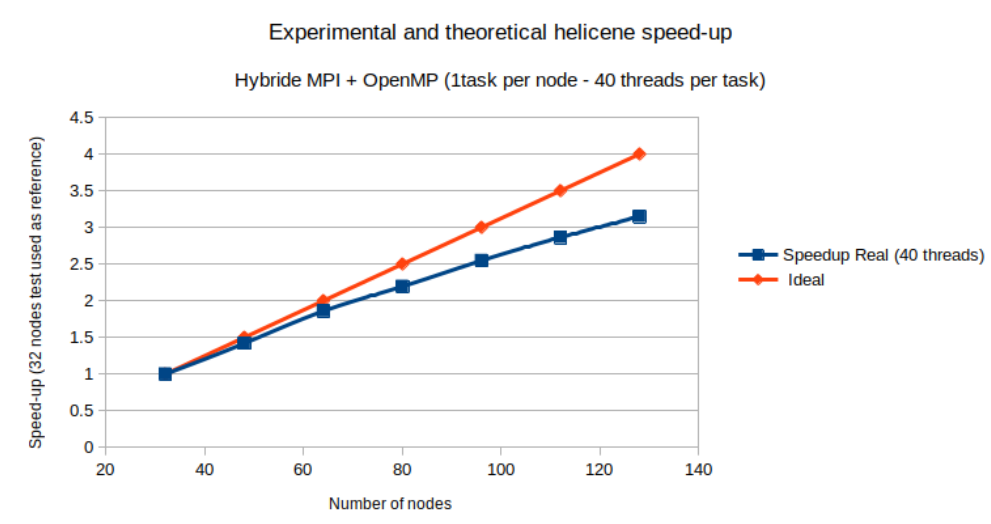


Figure 11: Scalability of the helicene test case. One node has 2 CascadeLake processors with 20 cores each. The reference runs on 32 nodes.

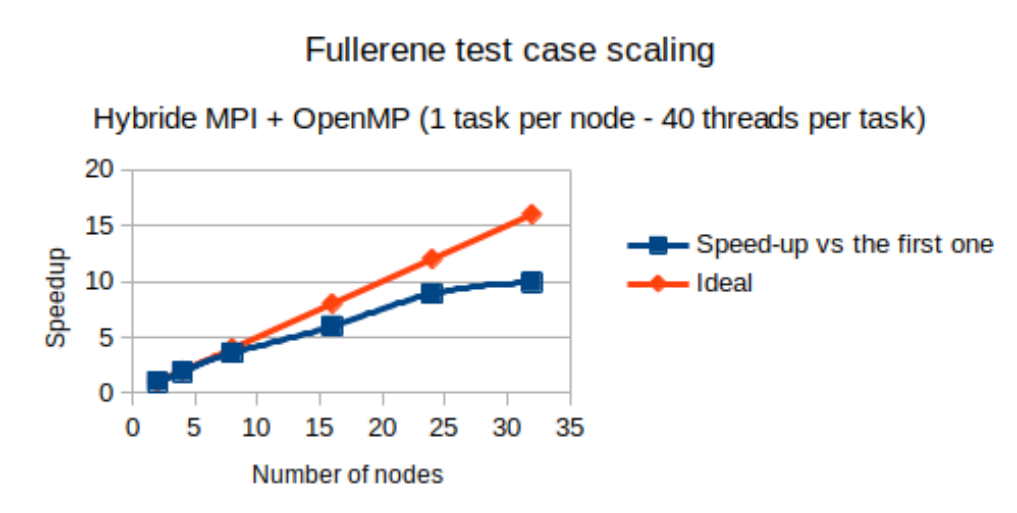


Figure 12: Scalability of the fullerene test case. One node has 2 CascadeLake processors with 20 cores each. The reference runs on 1 node.

Accomplished work:
The project tried to point out the possible bottlenecks inside the code. For that two profiling tools were used:

- Intel Vtune
- ARM Forge MAP

The profiling was performed with the following settings:

- Test case: helicene (large system)
- Number of nodes: 32
- Tasks per node: 1
- OpenMP threads per node: 40

Main results:
DDT "OpenMP overhead"

From the profiling the project was able to find some of the possible bottlenecks in the code. The main part of the time is spent in OpenMP overhead. This is mainly due to a part of the code, which generates OpenMP explicit tasks. Those tasks have to enter a critical section and therefore "wait". Another interpretation is that the profiler counts the time spent in the single section as overhead without taking into account the fact that the threads are working with the tasks.

From what the project was able to see, the algorithm used by the developers needs this part and they were not able to find a suitable solution to avoid it. One of the possible ways is to change the algorithm. But that it is a complex (if possible) task.

The expert performed several tests to see if the number of OpenMP threads have an impact on the performance. It was observed that the code runs slightly faster when only 20 threads per task are used. This might be due to a better memory usage (bus, caches, …) between the threads or due to the reduction of the overhead since less threads are generated.

MPI Broadcasting

Another possible bottleneck is the MPI broadcasting that covers between 10 and 15% of the time. From the analysis, one possible reason was a loop which broadcast the values of an array one by one. The project tried to send this array as one block but this decreased the performance. Therefore there is room for improvement.

2.9 Cut-off March 2020

2.9.1 Enhancing Parallelism in MAGMA2, 2010PA5263

Overview:
MAGMA2 is a recently developed Lagrangian hydrodynamics code for the simulation of compressible astrophysical flows. The code is intended to robustly simulate, for example, collisions between stars and tidal disruptions of stars by black holes and is intended to become one of the major "work horses" in our research group at the Astronomy Department at Stockholm University.

MAGMA2 is a smoothed-particle hydrodynamics code that benefits from a number of non-standard enhancements such as accurate gradient calculations that are obtained via the

inversions of small matrices, or reconstruction and slope-limiting techniques similar to what is used in finite volume methods. The calculations of densities and pressure gradients at each particle position require a loop over neighbouring particles. The enhanced accuracy of MAGMA2 comes in part from the use of high-order kernels which require large neighbour numbers (hundreds!). For an efficient neighbour search a fast "Recursive Coordinate Bisection" (RCB) binary tree has been developed. This tree is also used to calculate the by far most expensive part of most astrophysical simulations: the long-range (Newtonian) gravitational forces. Particular attention has been paid to make the calculation efficient, due to some algorithmic improvements our tree scales close to O(N), N being the particle number, rather than O(N log(N)) as standard tree codes.

The major goal of this project was to drive the previous efforts further and parallelise the code using a hybrid MPI/OpenMP approach. The project investigated various MPI and optimisation strategies. It focus on the most time-consuming parts of the code, namely the RCB-tree and the summation over long neighbour lists to obtain the density and pressure gradients.

Scalability results:

The scalability tests were evaluated for a case with 9882 particles on an Intel based system, Tetralith. The system has 1674 compute nodes with each consisting of 32 Intel Xenon Gold 6130 CPU cores.

The project compared the execution times on the subroutine `calculate_gravity` in seconds for 10 time steps using different MPI ranks. The execution times reduces from 87.9 seconds on a single CPU core to 7.49 seconds on 128 CPU cores, see Figure 13. A maximum speed-up of 11.7 was be obtained using 128 MPI ranks (cores) on 4 computer nodes as shown in Figure 14. However, the execution time increases when using 256 MPI ranks due to lower number of particles per MPI-rank, i.e. 38 particles/MPI-rank. The MPI point-to-point communication in the code can be improved by employing non-blocking or one-side communication, or a hybrid MPI-PGAS implementation.

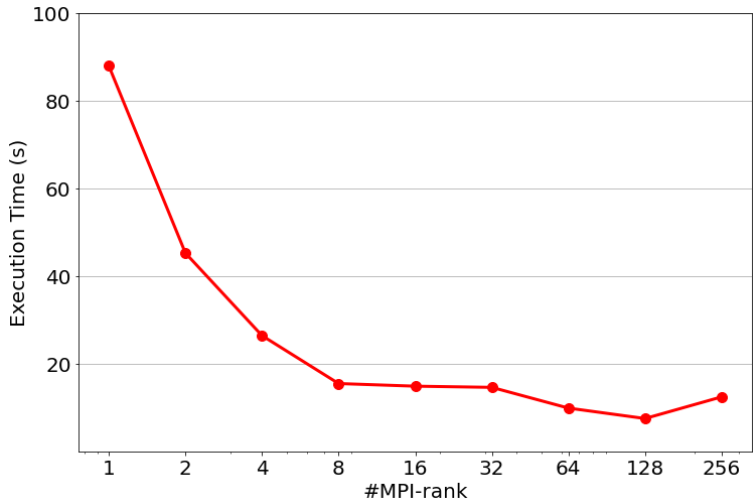


Figure 13: Execution time of calculate_gravity subroutine of MAGMA2.

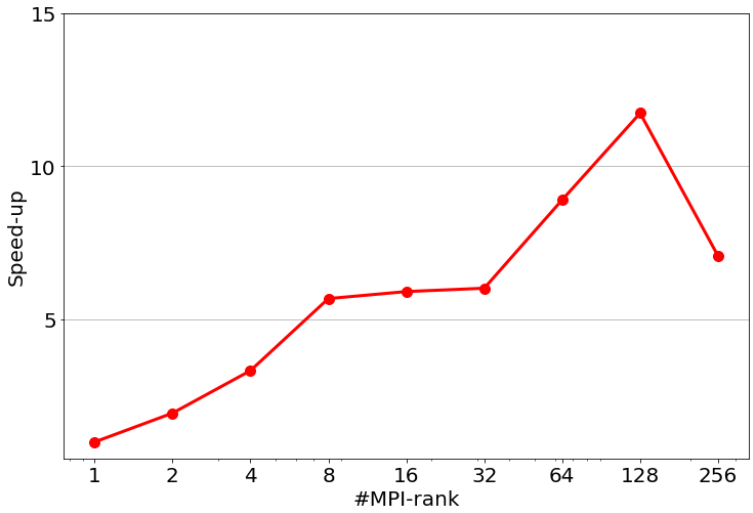


Figure 14: Speed-up of calculate_gravity subroutine of MAGMA2.

Accomplished work:

The project performed:

- Identification of the bottleneck of the code using profiling tools Allinea Map [8] and Score-P [10]. Allinea MAP is a parallel profiler with a simple graphical user interface to display the collected performance data. One screenshot for the profiling results for the serial code shows in Figure 15 below. It is clear that the subroutine `calculate_gravity` is the most time-consumption part and takes around 55.3% of total execution time. This subroutine `calculate_gravity` can be also identified using another profiling tool Score-P with Cube GUI interface, see Figure 16 below.

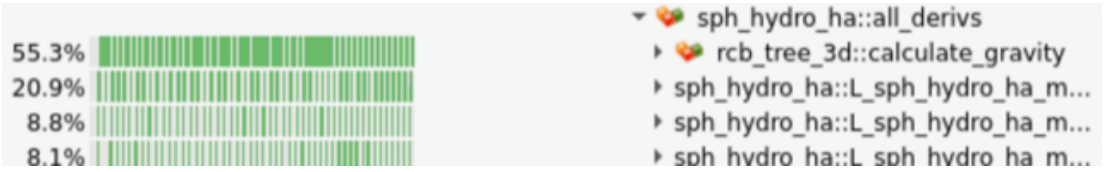


Figure 15: Allinea MAP screenshot of MAGMA2 profiling.

Figure 16: Cube screenshot of MAGMA2 Score-P profiling.

- Parallelisation using the MPI standard. Based on the profiling analysis, we first implemented MPI for the subroutine `calculate_gravity` using MPI `send/recv` point-point communications. However, the main algorithm used in the code is to create a fast tree for neighbour search and gravity calculations, i.e. it requires to go through all ll-cells (leukemia-lymphoma) to find the neighbour cells and their particles. The standard MPI communication is very expansive and takes much time due to the MPI latency over all ll-cells. One optimisation with MPI could be implemented by using only a few `MPI_Allreduce()` calls.
- Initialised implementations of global tree using MPI persistent communication. The project created a global tree including the MPI information over the local tree by using the MPI persistent communication introduced in [12]. The persistent communication can reduce the MPI latency but it requires advanced data structures in Fortran. That means that the Fortran code should be refactored.

Main results:
The PA project identified the calculation of gravity as a bottleneck in the smoothed-particle hydrodynamic (SPH) simulations that need to be improved. The PRACE expert implemented MPI for the most time-consumption subroutine and conducted the strong scalability tests for one case with 9882 particles. It was also illustrated to create a global tree using MPI persistent communication, which can potentially reduce the MPI latency.

2.9.2 XDEM4HPC: eXtended Discrete Element Method for High-Performance Computing, 2010PA5067

Overview:
The eXtended Discrete Element Method (XDEM) is an extension of the regular Discrete Element Method (DEM) with the additional feature that both micro and macroscopic observables can be computed simultaneously by coupling different time and length scales. Thus, this software fell in the category of multi-scale/multi-physics applications which can be used in realistic simulations such as combustion of materials and drug design. The different multi-physics components in XDEM are organised in a modular layout, i.e. conversion, dynamics, and computational fluid dynamics modules.

XDEM has already a good support for High-Performance Computing architectures, where it has shown to scale with more than 500 cores. The high level of parallelism in XDEM is achieved by using several paradigms such as, distributed memory (MPI), shared memory (OpenMP) and a hybrid of them.

In the present project, the PRACE expert targeted the OpenMP implementation which is one of the major bottlenecks in a typical simulation especially for the conversion module. In order to achieve this target, several steps were proposed:

- Initial profiling analysis of the OpenMP implementation to identify the bottlenecks. The profiling tools were Extrae/Paraver [9] and Intel VTune [11].
- Optimise the following modules in four phases
    - Optimisation of conversion module (phase I)
    - Optimisation of dynamic module (phase II)
    - Optimisation of conversion and dynamic modules (phase III)
    - Optimisation of conversion, dynamic and CFD modules (phase IV)

Scalability results:

In the present work we studied a biomass 3D example (see Figure 17). This example is relevant due to the global warming of our planet which is pushing us to find other sources of renewable and alternative energy. Biomass as a renewable carbon-based energy source is a sustainable alternative for generating power and therefore continues to grow in popularity to reduce fossil fuel consumption for environmental and economic benefits. Numerical simulations are therefore used in order to anticipate and improve the efficiency and optimisation of harmful gas emission.

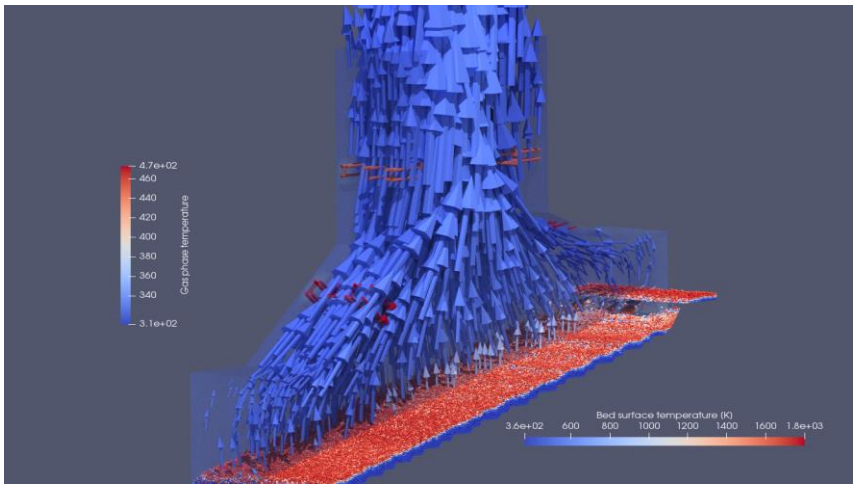


Figure 17: Biomass3D test case

The model we propose to predict is the entire biomass process, which is very challenging because it involves multi-scale, multi-phase, and multi-species phenomena including bed motion, turbulence, chemical reactions, and heat radiation. The fuel bed behaviour including its motion and different conversions are solved with XDEM (XDEM Dynamics and XDEM Conversion). The interaction of the fuel bed with the surrounding air is then taking into account through a CFD approach with the OpenFOAM software. This model contains 350,000 particles.

For the present study, only the OpenMP threaded version of XDEM was benchmarked.

Dynamics Module

For this module 1000 steps of simulation were done. The times for the runs are plotted in Figure 18 for varying number of cores, the speedups w.r.t. the corresponding 1 core simulation time for each branch are plotted in Figure 19. A slight performance gain was observed for this module. A higher performance for this module would require a larger data structure reorganisation.

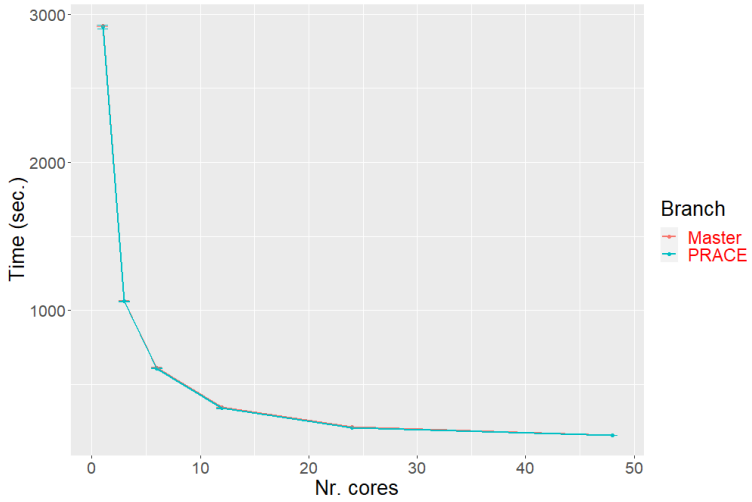


Figure 18: Timing for the Dynamics module as a function of the number of cores.

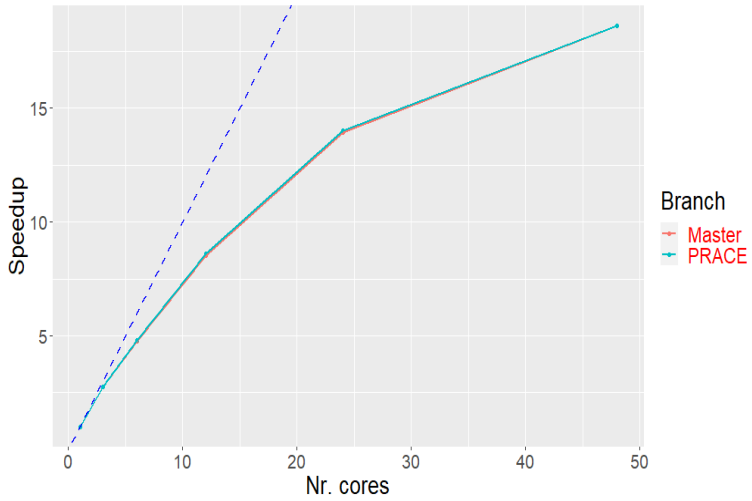


Figure 19: Speedup for the Dynamics module as a function of the number of cores.

Conversion Module

For the conversion module 100 steps of simulation were done. The times for the runs are plotted in Figure 20, the speedups w.r.t. the corresponding 1 core simulation time for each branch in Figure 21 a better performance gain was observed here because this module involves more functions that do actual computations.

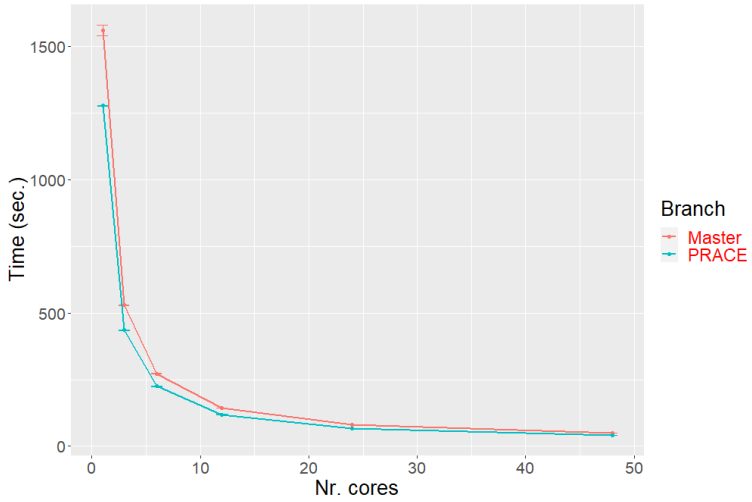


Figure 20: Timing for the Conversion module as a function of the number of cores.

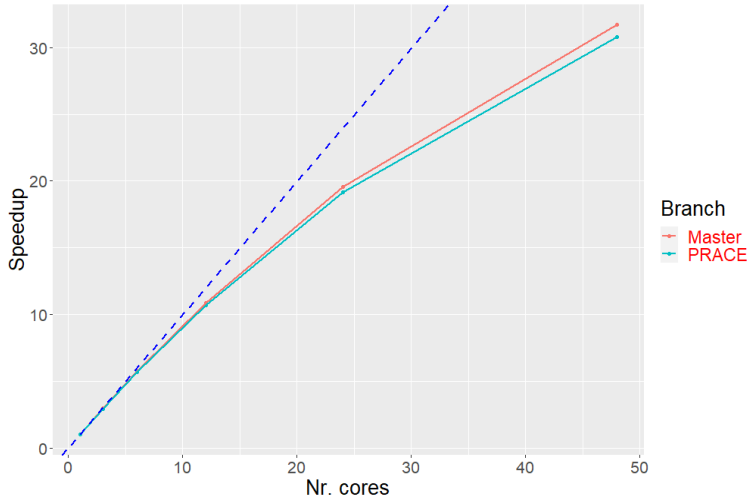


Figure 21: Speedup for the Conversion module as a function of the number of cores.

Dynamics-Conversion Modules

For these coupled modules 100 steps of simulation were done. The times for the runs are plotted in Figure 22, the speedups w.r.t. the corresponding 1 core simulation time for each branch in Figure 23. The performance gain observed here was similar to the one of the pure conversion module.

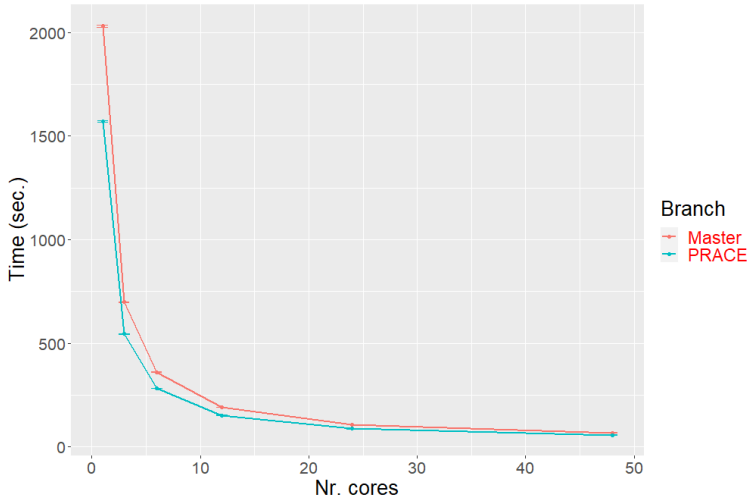


Figure 22: Timing for the Dynamics-Conversion modules as a function of the number of cores.

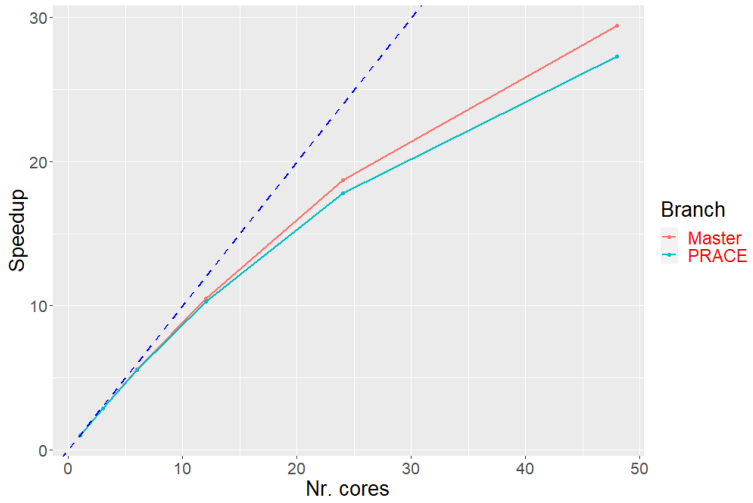


Figure 23: Speedup for the Dynamics-Conversion modules as a function of the number of cores.

Accomplished work:

Several optimisations were done in the present project. We used Intel VTune (v. 2019) and Extrae (v. 3.7.1) profiling tools to investigate the performance of the most expensive routines.

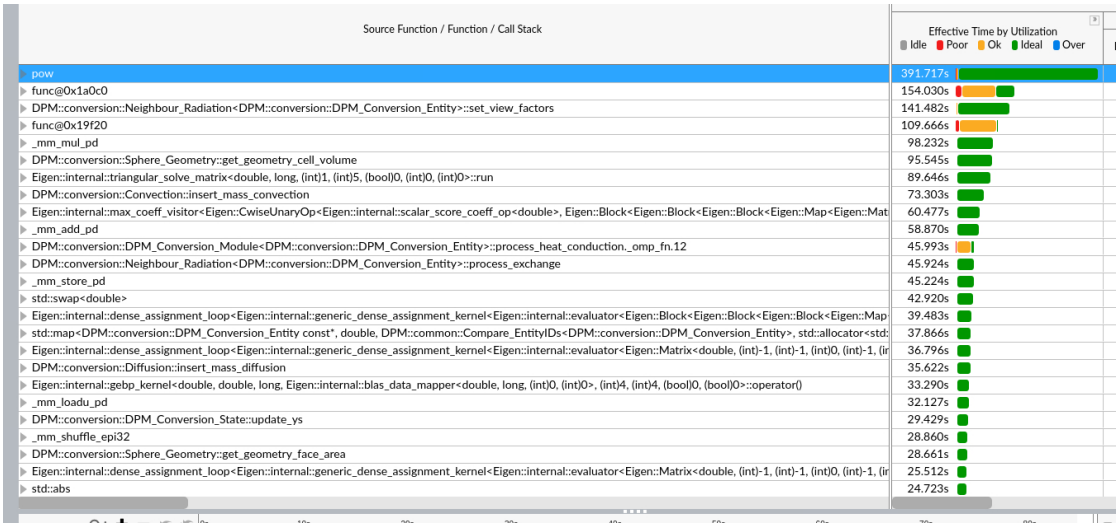


Figure 24: Initial profiling analysis with Intel VTune showing the most expensive function calls.

The initial profiling analysis (see Figure 24) we obtained from VTune showed that the function `pow()` is being called many times and it was the major contributor (~14% of the total time). We inspected the code and detected that this function was used to compute products such as $pow(x, b) = x^b$ (with b integer), in this case, we proposed to use explicit multiplication if b=2, and for b >2 we suggested to use additional auxiliary variables to compute partial products. Arithmetic operations with a fractional value of b where substituted with integer operations.

Other optimisation strategies we performed:

- Exponential expressions such as the Arrhenius factor were transformed to logarithmic expressions. We observed a significant reduction in the total simulation time by implementing this modification (~10% of the total time).
- Creation and initialisation of structures was taken out of loops when possible, as an example in the function `set_view_factors()`.
- We observed that some checking functions such as `DPM_ERROR()` slowed down the code and some of them are now used only in the debug version `DPM_ASSERT_OR_ABORT_DEBUG()`.
- In function `process_heat_conduction()`, an overhead was detected which was traced back to the opening of two consecutive parallel regions. We rearranged both regions so that they could fit into a single parallel region where data initialisation, without dependencies, was performed outside the parallel region.
- Some functions like `update_ys()`, made use of complex loops definitions, we transformed these loops into lightweight loops by defining the iterators outside them.
- The functions `get_grid_size()`, `cell_grid.size()`, `cs_vec.size()` were called many times, for instance in the function `broadphase()`, in the initial version and they represented about 1% of the total time, taken together. In order to avoid these calls, we defined auxiliary variables and saved the value of the `size()` functions instead of computing them explicitly. A similar approach was taken in other parts of the code, for instance in the function `process_exchange()`, where functions such as `get_particle_surface_shell_rho_cp()` were called several times inside loops. We noticed that some function calls can be avoided by introducing additional auxiliary variables. However, the choice of auxiliary variables could make the code less easy to interpret. This could be solved however with an appropriate choice of names for the auxiliary variables. Two of the functions that benefited by the use of auxiliary variables were `insert_heat_conduction()` and `insert_mass_convection()`.

Main results:

The conversion module and the coupled dynamics-conversion modules were sped up by ~16% (48 cores). The percentages of the total time for the most expensive loop modules for the master and PRACE branches are the following (48 cores):

Module	Master	PRACE
Broad phase	0.32	0.33
Narrow phase	0.13	0.15
Apply bed surface	0.59	0.64
Apply heat conduction	4.24	4.51
Apply heat radiation	2.95	2.07
Integration conversion	66.38	58.88

Table 2: Performance evaluation of the different modules of XDEM for 48 cores and for both Master and PRACE branches (in percentages %).

The project also published a white paper, which can be found online under [13].

2.9.3 Scalability of Automatic Design Optimzation Algorithms, 2010PA5280

Overview:
The most important type of cases executed at Diabatix is a design run. Such a design run fully determines the most optimal design for a specific cooling problem. A design run is an iterative process, with each iteration being executed comprising a full CFD simulation run. Apart from the CFD solver run at the middle of a design iteration, an important part of the workflow is handling the design. The "handling" is mostly operations of fields, scaling in size with the number of design variables. The current handling relies on a significant amount of I/O operations. The project aimed at the reduction/removal of the overhead resulting from I/O.

Scalability results:
Benchmarking results are presented on the Figure 25 and Figure 26. Both figures show the gain of using the process developed within the PRACE project, relative to the original process sequence. The gain is dependent on the size of the case being executed, the number and type of nodes the case is executed on and the number of time steps needed to perform one design iteration. As the number of time steps is normally dependent on convergence, this number was fixed for the purpose of the tests, with the goal of a more consistent comparison.

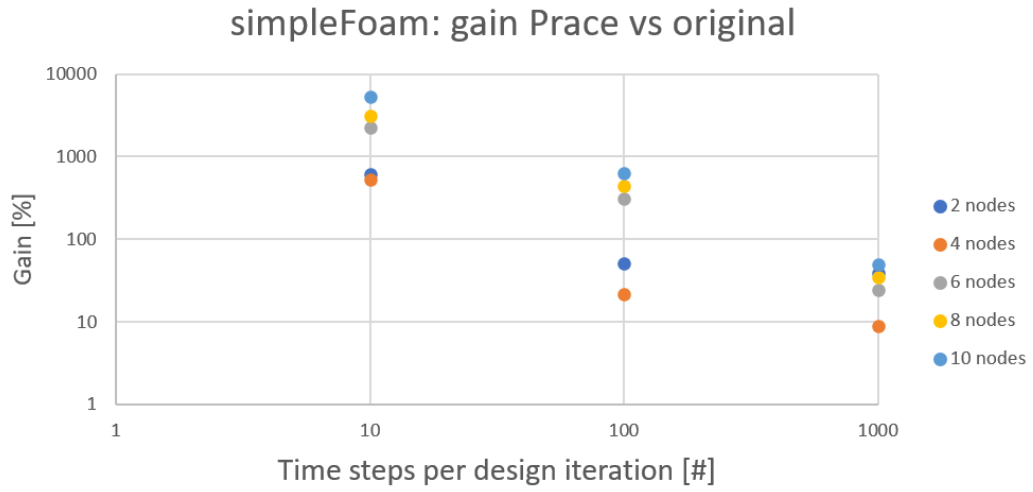


Figure 25: The gain in process time for the process sequence as per the PRACE project relative to the original process sequence for the simpleFoam solver. The test case is pitzDaily with 12 M cells executed on VSC Breniac's Broadwell nodes.

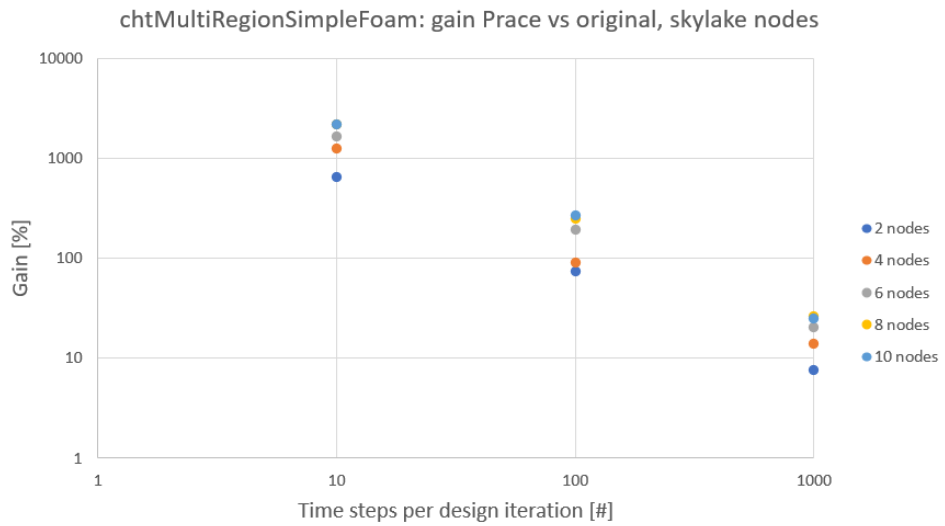


Figure 26: The performance gain for the process sequence as per the PRACE project relative to the original process sequence for the chtMultiRegionFoam solver. The test case has 4 M cells in the fluid and 2 M cells in the solid, executed on VSC Breniac's Skylake nodes.

In general, the new sequence shows a significant gain for all situations. The gain is higher for lower number of time steps and for higher number of nodes. The results can be further explained by the following observations:

- The I/O-related times (decomposition, reconstruction and reading/writing) are almost identical for a specific number of nodes.
- For the original sequence, the I/O-related times go up when using more nodes. For the PRACE sequence, these times are almost independent of the number of nodes.
- The solver-related times show a slight absolute difference in favour for the PRACE sequence. Relatively, this has the largest effect when using a small number of times steps. This is the advantage of avoiding the part of the solver where files have to be re-read and loaded into memory.

When using Skylake nodes vs Broadwell nodes, the gains are slightly lower. From separate scaling tests, Skylake nodes can be expected to be 40% faster in general, but the gain is higher for the serial decomposition and reconstruction times. Hence, decomposition and reconstruction takes less time in the original process sequence, leading to smaller gains for the PRACE project sequence.

The range of the time steps chosen for the test is on the low side, with 1000 time steps per design iteration actually being more realistic and leading to gains of 10 – 60%. Nevertheless, lower time steps were included for:

1) Showing the consistency of the decomposition, reconstruction and reading times over all cases.
2) Near the end of design run, the number of iterations will typically decrease.

Also, the results are consistent enough to make extrapolations for even higher time steps. When applying these results to a realistic case being processed on Skylake nodes with 5k time steps per design iteration, the gain is around 5%.

Accomplished work:

At the start of the project the communication of information happens through rather expensive I/O operations because each design iteration contains an OpenFOAM solver run. The solver is an executable that writes the outcome to disk before it exits. This process is depicted in the left flowchart of Figure 27. The goal of the project is to transform the solver in a way that it does not exit, keeping the outcome in memory and not dumping it to disk. This enables the "update design field" step to perform its work in memory, effectively removing I/O from the design loop. This process is depicted in the right flowchart of Figure 27.

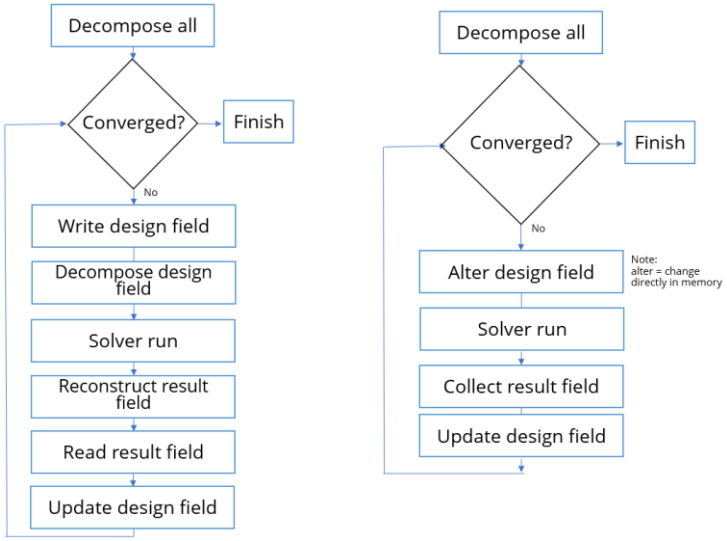


Figure 27: Flow chart of a design run. Left figure: current, I/O intensive workflow. Right figure: envisioned in-memory workflow.

Since the design loop itself is implemented in Python, the following approach was proposed and implemented successfully. The `main()` function (C++) of the OpenFOAM solver which comprises three sections: Initialisation of the simulation, the simulation run itself, and the cleanup sections where the results are written to disk. This function is transformed to a C++ Solver class where the initialisation of the simulation is performed in the Solver class constructor, the simulation run is performed in a `loop()` member function, and the cleanup in the Solver class destructor. In addition, some functionality is added to access all the fields in the simulation. This Solver class is exposed to Python as a Python binary extension module, called pyFoam. The process of converting C++ code into Python binary extension modules is facilitated through Micc [14]. With pyFoam a Python script can setup a simulation, run the solver, access the fields computed by the solver run (in memory), modify the design fields (in memory) and run the solver again. These steps are repeated unit the design run completes. The need for I/O communication is completely removed.

At the same time the overhead of I/O operation in the Diabatix solvers was improved using binary reading & writing as well as using the host-collated file format.

Main results:

Technical conclusion:

With the aid of Micc a complex C++ code was turned into a Python binary extension module, that could completely eliminate the I/O overhead and which provides a user friendly, intuitive and flexible Python class that is far easier to manipulate than a loop over a series of executables communicating with each other through I/O.

Practical conclusion:

Shifting priorities due to short term perspectives sometimes complicate the implementation of successful solutions when working with companies.

The project also published a white paper, which can be found online under [15].

2.10 Cut-off June 2020

2.10.1 HOVE3 Higher-Order finite-Volume unstructured code Enhancement for compressible turbulent flows, 2010PA5404

Overview:
HOVE3 is the follow-up project of HOVE and HOVE2 and focusses on the optimisation of the code UCNS3D, which is a CFD solver for simulation of compressible turbulent flows.

Previous developments of the UCNS3D code under the support of PRACE included optimisation of implementation through the use of linear algebra libraries, code restructuring, and inclusion of MPI-IO operations that also enabled very large grids to be used.

The target for this project was to find out the optimum configuration of combined OpenMP and MPI configurations to run the UCNS3D code on the new AMD EPYC based HLRS-HAWK machine.

Scalability results:
In order to optimise the OpenMP configuration, various tests were compared and the scaling behaviour is shown in Figure 28.

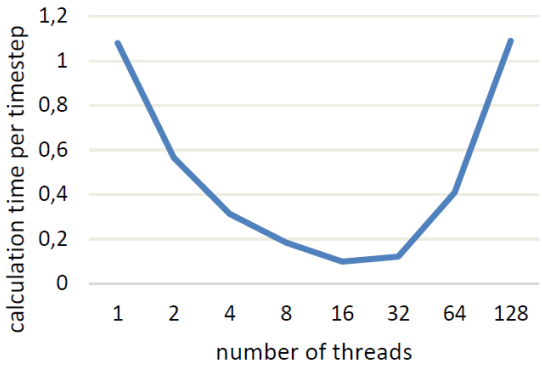


Figure 28: OpenMP scaling behaviour of UCNS3D

All runs were performed on two nodes with variation of the number of threads. For 8 to 32 threads good scaling is shown.

Analysing the performance for 16 threads, an optimum of MPI processes is visible for 16 processes.

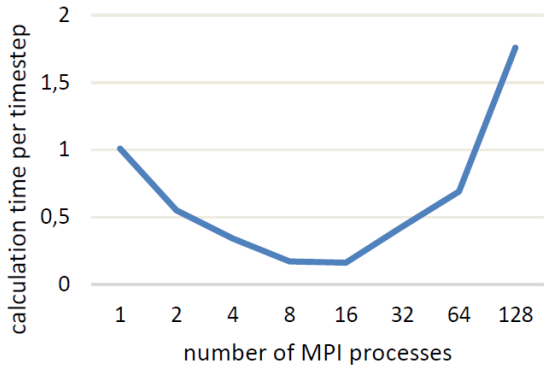


Figure 29: MPI scaling behaviour of UCNS3D

Accomplished work:
The comparison of performance with GCC and Intel compiler shows better performance for Intel.

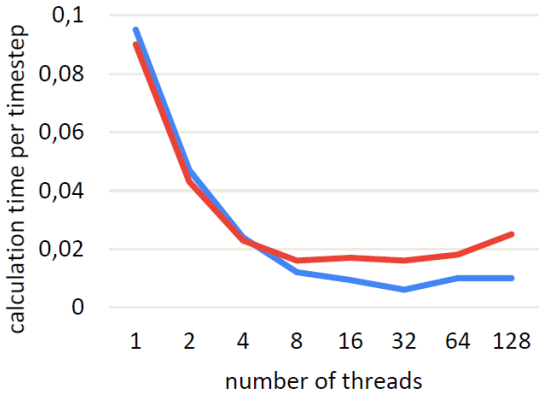


Figure 30: OpenMP scaling behaviour of UCNS3D in comparison of the GCC (red) and Intel (blue) compiler.

The generated outputs of the simulation showed incorrect results for some configurations with GCC. Changing the algebraic libraries and varying compiler flags did not influence this issue. Since this problem did not occur on other architectures, further analysis is needed to clear the issue.

Main results:
UCNS3D showed good strong scaling. The optimal configuration on Hawk was found at 16 threads and 8 or 16 processes. For further scaling tests on HAWK the Intel compiler and MKL libraries should be preferred.

2.11 Cut-off September 2020

2.11.1 Use of HPC for optimisation of drinking water distribution networks, 2010PA5511

Overview:
The simulation code (Gondwana) in this project helps with optimisation of a number of drinking water distribution networks, comprising the optimisation of pipe diameters, and the location of valves, pressure and flow sensor locations. The networks in question have a variable size (model sizes from 5000 to 50000 nodes), require the simulation of up to 500 scenarios and the

computation of ca. 10^8 function evaluations. This would lead to a computational time of ca. 10^6 hours (on one single CPU). In this context, the PI would like to explore the use of high-performance computing in the optimisation of drinking water distribution networks. Shortening computational times in these types of problems would allow to better serve the water sector, by providing answers considering a much wider range of relevant scenarios in a more feasible time and being able to run multiple optimisation problems at the same time.

Scalability results:

The simulation code (Gondwana) uses shared memory and the multiprocessing python module. We execute it on one RHEL 7.8 Intel server (dual-socket E5-2690v3) from two to twenty four processes.

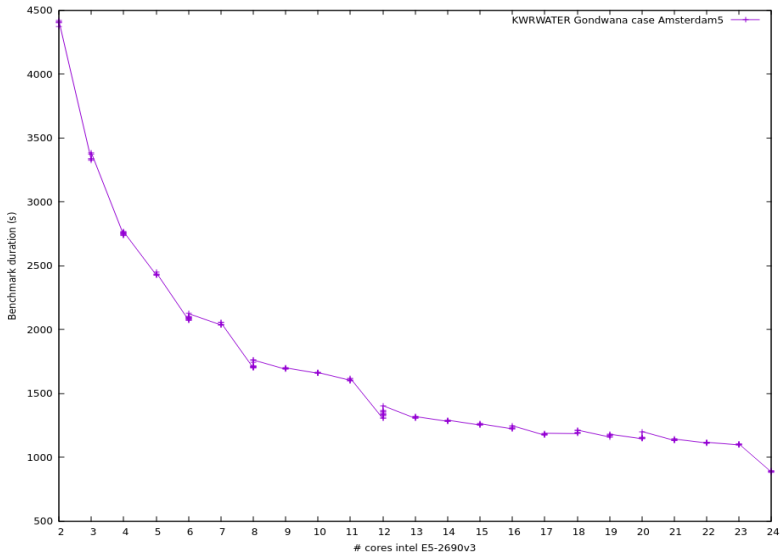


Figure 31: Gondwana scalability behaviour

Accomplished work:

- Porting by removing the need of a GUI on the compute servers.
- Find which Python modules needed by Gondwana are compatible with the `intelpython2019` module provided by the CINES computing center.
- At each restart of the simulation, keep the same seed for random numbers generator.
- We used `log-sparse` function of the viztracer profile and choose then to decorate Gondwana functions that we wish to trace. In the traces, we noticed that child processes from the multiprocessing python were created at each simulation iteration. We factorised the process creation by doing it once at the beginning of the simulation. We also pinned with `os.sched_affinity()` each Python subprocess created by the multiprocessing module to a unique and the same core across the iterations.

Main results:

Now Gondwana controls the way its parallelism is distributed among the cores of a compute node and the benchmark timings are more predictable thanks to the fixed seed of the random number generation. Thanks to these improvements we plan to use a big Skylake compute node with 224 cores for solving an industrial testcase.

2.11.2 FESOM2 Finite volumE Sea ice Ocean Model enhancement, 2010PA5513

Overview:
FESOM2 is a next generation global sea ice-ocean model that uses unstructured meshes, developed at the Alfred Wegener Institute for Polar and Marine Research. It is used for regional and global studies in stand-alone mode as well as in the coupled settings with several atmospheric, biogeochemical and glacier models. FESOM2 is an ocean part of the AWI-CM climate model that participates in CMIP6 [20] and recently was made available as an option in the European community Earth-System Model (EC-Earth).

In order to try to improve the FESOM2 performance, this project was used as a first collaboration between AWI and BSC for a computational profiling analysis of FESOM2. The BSC has long experience of working with earth system science codes and in particular with ocean models. The performance tools, developed at BSC, can potentially provide very valuable information about model behaviour in the supercomputer environment and help to identify the ways for further model improvements and optimisations.

Scalability results:
The computational analysis was done using the profiling tools known as BSC Tools (Extrae and Paraver [9]). To make this possible, FESOM was deployed on Marenostrum4 using a global commodity resolution (Core2) and a standard set up supervised by developers of the model. Additionally, debug options were added to be able to evaluate different aspects such as user function calls, needed by the instrumentalisation.

The first step in a profiling analysis is to have a general overview of the trace application structure. Figure 32 shows the MPI call type duration of an execution of FESOM of 20 time-steps, using 144 MPI processes. The complete execution was divided into 3 steps: a first step of initialisation with several broadcasts (Init), the second step of initialisation with broadcasts and synchronised communication (Init2) and the time step execution including 20 time steps (Iterative process). Since these models are used for climate prediction, the iterative process usually is much longer than the initialisation, which happens only one time at the beginning and usually represents no more than 1-2% of a complete execution.

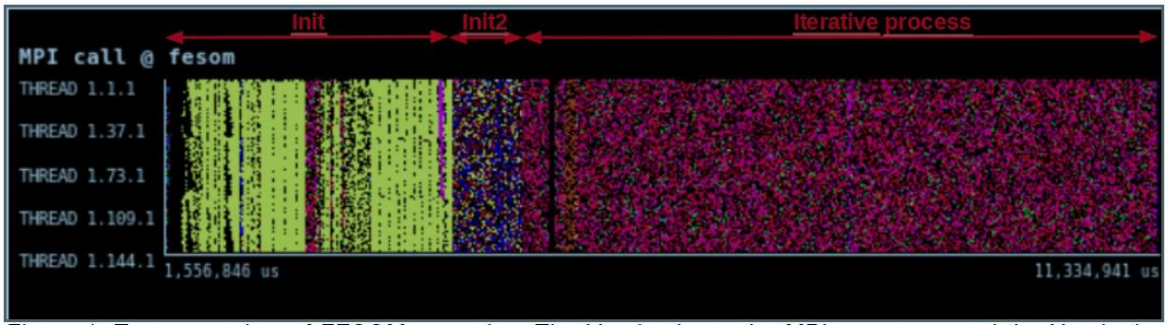


Figure 32: Trace overview of FESOM execution. The Y axis shows the MPI processes and the X axis the execution of each process along the time. The colours show different MPI events.

On the other hand, with the iterative process containing 20 similar time steps, the analysis focuses on one representative time step to reduce the size of the trace, taking into account that all time steps are similar. Additionally, each time step can be divided in two different areas for the physical processes calculated in each time step. Figure 33 shows a new trace including only

the execution of one time step where the two areas explained are separated for the ice and ocean calculations and using 144 MPI processes.

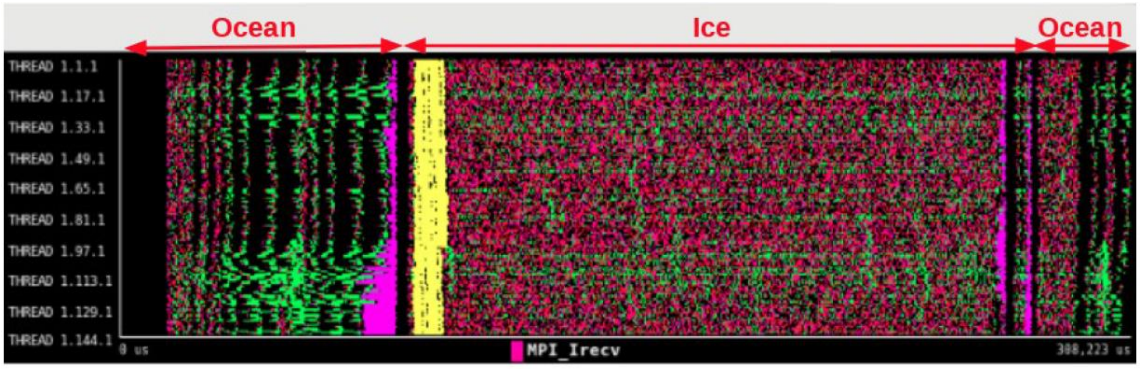


Figure 33: Trace overview of one regular time step of FESOM. Ice and ocean phases are highlighted.

It is important to highlight that the ice phase seems to represent more than 50% of the time step execution. However, it was proven during the project that it was an instrumentalisation problem of Extrae. The reason is the more than 200 MPI communications inside this area, which produces an extra overhead of Extrae to collect all the events. This issue was solved collecting traces without the callers. However, the analysis can be followed using these traces since total times were not used for the study (pursuing the main possible bottlenecks of FESOM).

The profiling analysis was focused on the scalability of Init2 and one time step of the Iterative process. A strong scalability test of each area was done. This test helps to evaluate the fundamental aspects of what performance is in parallel applications. In particular, three values for the number of MPI ranks were used (144, 288 and 432 MPI processes). Figure 34 shows a general overview of the strong scalability test. It is important to highlight that the intrumentialisation problem found in the ice phase is affecting the scalability of the time step execution in Figure 34. This means for example that the execution using 288 MPI processes should be two times faster than using 144, which is not appreciated in Figure 34 due to the instrumentalisation.

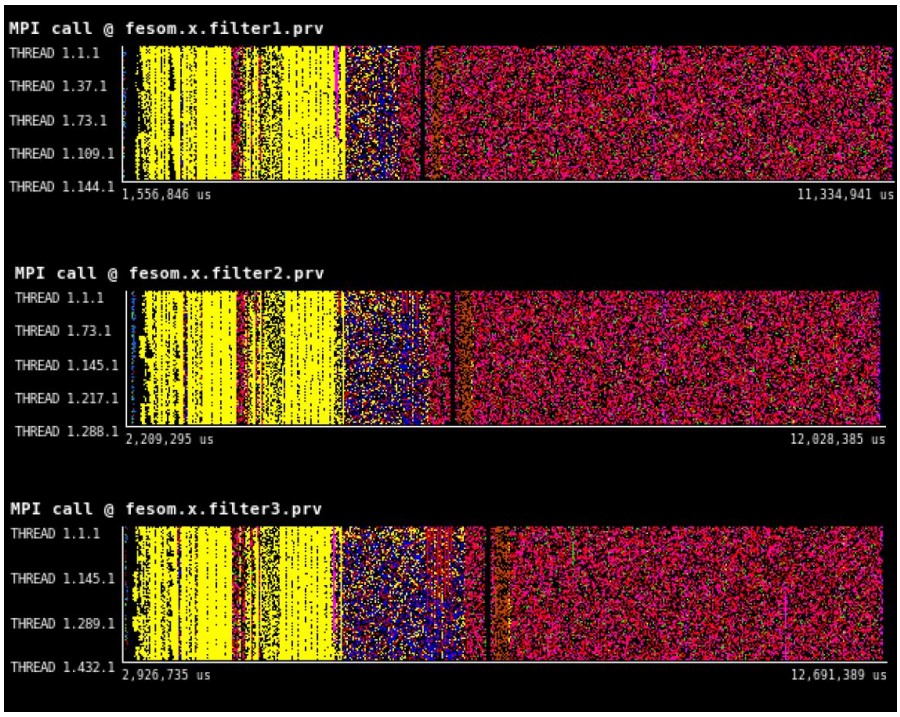


Figure 34: General overview of FESOM strong scalability test for 144, 288 and 432 MPI processes respectively.

Accomplished work:

Taking into account the complexity of FESOM as a complete ocean model and that this is the first interaction between BSC and FESOM developers (AWI), the accomplished work and the resources were used for the profiling analysis itself.

This profiling analysis was focused in one part of the initialisation (Init2) and the iterative process of FESOM through one regular time step. The analysis evaluated:

1) The overhead of MPI as parallel application.
2) The computational performance through hardware counters (PAPI). In particular, the instructions per cycle (IPC), the memory locality (cache misses per 1000 instructions of L1, L2 and L3), the vectorisation of the model according to the floating point operations computed (VEC) and the parallelism of each event (number of processes doing real computation in parallel for each phase).

From the profiling analysis done, some issues were found according to the MPI study:

1) There is a collective communication in the time step trace which represents close to 5% of the total execution time. This phase is not scalable using more parallel processes. The main reason is the irregular pattern of communication among different MPI processes, producing a load imbalance which delay the computation of the next phase of some of the processes.
2) The ice phase contains more than 200 point-to-point communications (isend+irecv+MPI_WaitAll). This produces a very low granularity in the computational side among communications. Additionally, there is some load imbalance in the computation of the different subdomains (as it can be seen at the next point). All these reasons produce an important bottleneck in the MPI communications due to the latency overhead between consecutive communications. Moreover, the low granularity plus the communications produce a serialisation of the operations where the asynchronous

communications are not really working. Due to these issues, this phase cannot scale when more MPI resources are used.

3) The useful duration histogram of the ocean first phase in the time step trace shows load imbalance among different MPI processes. This is due to the domain decomposition algorithm used. Figure 35 shows that some subdomains contain more layers than others (when the vertical space is computed). This produces that some MPI processes finish their work before others. Since there are some synchonisation points along the ocean calculations (due to collective communications as reductions or point-to-point communications), the cost of the load imbalance produces an extra overhead of the parallel implementation, which reduce the computational performance of FESOM in the general during the computation phases.
4) In the Init2 area section the overhead increases with the number of MPI processes used. The irregularities in the communications produces a serialisation which makes the increment of the final execution time. Though it is outside the iterative process, Init2 is not scalable and the more MPI processes are used, the more overhead will be produced.

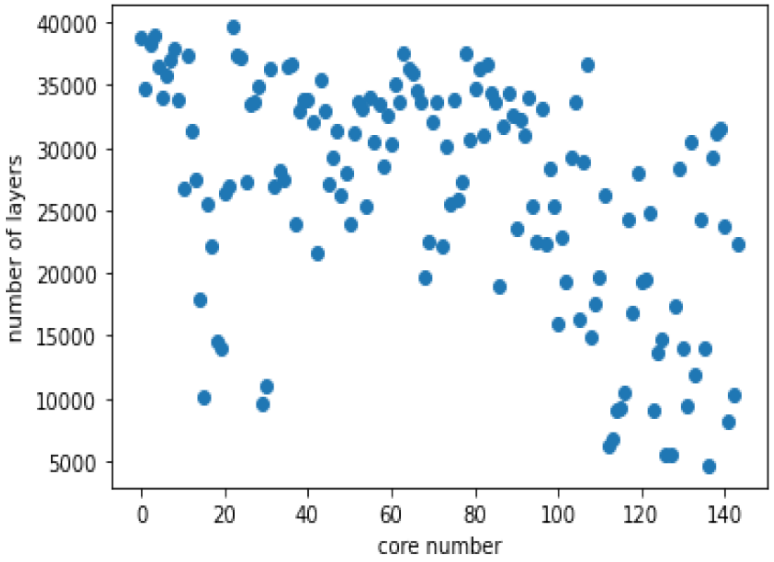


Figure 35: Relation between the total number of layers of each subdomain and the MPI process (core number), using 144 MPI processes.

Additionally, some issues were found during the PAPI counters study:

1) The instantaneous parallelism of the time step phase on average is 36% for the 3 MPI traces tested. This means that in average only a 36% of the MPI parallel resources are doing useful work (for example, they are not waiting for a communication).
2) The low parallelism showed in the previous issue reveals that the basic PAPI counters studied (IPC, LX cache misses and VEC) could be affected with a low performance since many of the parallel resources are waiting for other MPI processes in synchronisation points instead of doing real computation. The useful IPC is in general lower than IPC, proving that the low parallelism affects the performance of the execution. Moreover, the ice performance is affected dramatically, while the ocean areas are affected close to the communications, due to the load imbalance as it was explained before. Similar results were obtained for cache misses and vectorisation. For this reason, useful configuration was applied to IPC, LX cache misses and VEC

configurations to check the real performance of the processes doing calculations, obtaining the following conclusions:

- Two areas of the ocean calculation have low IPC, correlated to high cache misses in L3, L2 and L1 memories.
- One area of ocean calculation has high IPC, low cache misses but a poor performance of the vectorisation. The high IPC could be correlated to the low vectorisation.

Main results:
The main goal of this project was the computational analysis itself, looking for possible bottlenecks and suggesting possible optimisations for the future. The recommended optimisations could not be the only possible solution. The outcome can be used in the future to introduce the optimisations, find new solutions or evaluate the possible solutions. Additionally, a new computational analysis can be applied to compare the model performance studied with the new performance model with some of the optimisations introduced.

According to the results provided in previous sections, the main issues and possible solutions are listed below:

- Review broadcast communication to avoid the irregular pattern which produces a load imbalance in the communications.
- Review initialisation algorithm to avoid the irregular pattern which produces a load imbalance in the communications.
- Evaluate the possibility of OpenMP to avoid serialisation.
- Evaluate the possibility of the reduction the MPI communications during the ice calculation if OpenMP is not a possibility.
- Review the domain decomposition algorithm to improve the load balance in the ocean calculations.
- Review the locality of the two areas with low IPC and high cache misses to improve the performance of a computation phase.
- Review the dependencies in the calculation loops for the area of calculation with low VEC.

2.11.3 Fast MDS, 2010PA5526

Overview:
The Metabarcoding application developed in collaboration between INRAE and INRIA in Bordeaux aims at characterising the biodiversity of an environmental sample through bioinformatics and molecular data processing. The diversity of an environmental sample is hidden in a set of N sequences (parts of the genome recognised as taxonomically relevant) which have been produced by NGS upstream (sequences of possibly all the microorganisms of it). The second step (which is computationally intensive too, and has been done upstream as well) is to compute all pairwise distances between sequences (edit distances). The objective is to reveal the structure of the distance matrix by running a multidimensional scaling (MDS) on it: that is, build a point cloud in an Euclidean space of sufficient dimension, where geometrical distances mirror genetic distances as much as possible. The critical part of the processing is to compute a singular value decomposition (SVD) of the N x N Gram matrix computed from the distance matrix. It is not feasible beyond a given limit for N. The objective of this project is to run MDS with a random projection algorithm, which enables to go as far as possible beyond

this limit. The role of MDS, by reducing the dimension of the problem, is to further enable methods based on supervised / unsupervised learning to analyse the point cloud (e.g. extracting clusters).

The aim of the project was to leverage INRIA Bordeaux's HPC software stack composed of the StarPU task-based runtime system and the Chameleon dense linear algebra solver, to achieve the distributed processing of million sequence samples on new architecture (AMD processors).

So far, the project applicant has focused on the optimisation and scalability of GEMM, QR and have put aside the reading of data. Also, the total time is dominated by reading in the full matrix. Depending on the size matrix this time represents more than 90% of the total time.

The targeted optimisation work was

- Optimise I/O and the way to read the data from a node by changing the size and the shape of the data block of the matrix to be read and/or using the burst buffers to preload the matrix.
- Improve GEMM algorithm

Scalability results:
We present the different numerical results and scalability. Our experimental results report the average times over 5 trials for each algorithm and configuration.

First, we present the results of our micro-benchmarks to study the behaviour of the AMD nodes:

Processor	Compiler	Threads:1	2	6	10	16	18
SkyLake (ref)	Intel2019	15.51	6.18	4.44	1.99	3.45	3.39
AMD	Intel2019	30.84	10.29	6.92	2.43	4.04	3.85
AMD	GCC	19.57	7.83	4.98	2.03	3.43	3.32

Table 3: Results in seconds on AMD of fastMDS for a matrix 20,000x2,000 - with MKL 2019 update 4

As the sequential part of the code is the SVD of a large and sparse matrix of size 99,594x9,959, which is the smallest size of the SVD we are processing. Since the number of threads on a node is large, Table 4 shows that we can use all the threads on a node for matrices of size greater than 100,000 rows.

Threads	32	64	96	128
Time (sec)	360.47	284.73	259.21	288.04

Table 4: Time to compute an SVD on a matrix of size 99,594x9,959

I/O optimisation

Here we present the result of the optimisation performed to read and construct the distance matrix on a matrix of size 99,594x99,594. Table 5 shows the time in seconds to read and redistribute the distance matrix. Only one thread per MPI process reads the value so as not to stress the disk access. Even on a single node, the new algorithm is faster than the old one.

MPI processes	Old algorithm	New algorithm
1	538	314
2	294	168
3	196	115

Table 5: Time in second to read and to fill the matrix

Scalability

We are now interested in the scalability of our approach for different matrix sizes (270,983, 426,548, 616,644, and 1,043,192). Figure 36 shows the computation time in seconds for reading/constructing the matrix and the full cost of the MDS depending on the number of MPI processes. Each process uses 123 threads. We notice that the reading time scales well depending on the number of processors. In this part of the code, we manage to reach the same reading time whatever the size of the matrix.

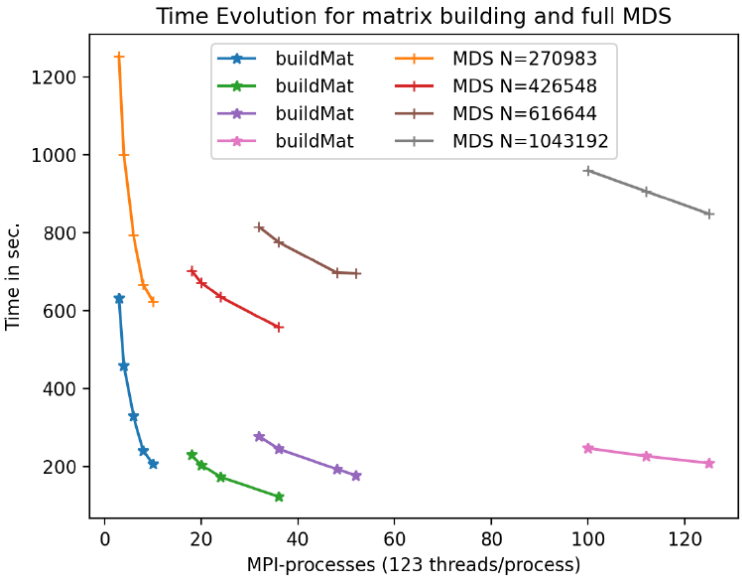


Figure 36: Scalability for different matrices.

Concerning the MDS part, the gap with the reading/building remains constant and around 300 seconds. This time corresponds approximately to the time of the SVD calculation which is multithreaded on a node. These results show the limit of our approach, it requires either to go to a parallel SVD (distributed) or by alternatives without SVD.

Asynchronous approach

The last result concerns the study of the asynchronous approach to see if we can overlap the reading of the matrix by calculations.

We have compared the CPU times with and without using the asynchronous interface and the results were not satisfactory, i.e. asynchronism not giving better performances. This can be explained by the fact that in our case we have to read a large matrix first (without computation before to be overlapped) which makes use of only one CPU (HDF5 not thread safe, other CPUs are waiting for tasks) and that just after the algorithm involved is a Gram matrix computation. This algorithm is not very costly; it involves sum squares accumulations then an update of all

single values of the matrix with sum squares accumulation column-wise, row-wise and element-wise, but it requires all values of the matrix to be loaded. Thus only this not costly algorithm can be interleaved with the reading tasks and cannot win lot of time here.

Accomplished work:

Preparatory work

Both the AMD processors and the number of threads available on a node (128 threads) were new for our application. Also, as the application uses mainly LAPACK, the first step (preliminary work) was to select the right compiler that gives the best times with the MKL on AMD. Then, we looked for the number of threads to use on the SVD according to the size of the matrix.

As a result of this study, the GCC-9 compiler was selected over the Intel compiler. We did not manage to have computation times with the Intel compiler close to those obtained on a platform with Intel processors. We have seen that for the target arrays, we can use all the threads of the node without performance loss.

We used our own software stack instead of utilising the preinstalled modules for the following software: hwloc, UCX, Open MPI, hdf5, and fxt. This approach allows us to preserve homogeneity in the results compared to different platforms.

The first results show that the I/O is the main problem.

I/O optimisations

1) The matrix is a matrix of blocks and each block or tile is 320 by 320 in size. The current version reads a block from the matrix from the hdf5 file. This block does not match the block used to generate the hdf5 file. We changed our strategy to read an hdf5 block from the data and redistribute it to the right processor and place it in its right location in the local structure of the matrix.
2) Our project work plan included experiments with Hawk's DDN Infinite Memory Engine (IME) burst buffer I/O system. However, due to stability issues of the IME, we kept using the regular file system instead.
3) Since the burst buffers are not available, one alternate idea is to study how we can remove the synchronisation between the reading step and the construction of the Gram matrix and the beginning of the random projection. The goal is to pipeline the first steps and then recover the reading by some computations. To do this study, we only focus on a single node. We have developed a new wrapper to handle the asynchronism of the Chameleon matrices and adapted the random projection to force synchronisation when necessary.

Main results:

Optimisation for I/O has been studied and has reduced the CPU time spent by a factor of 7.5 on our largest test case of 1 million degrees of freedom and over 50 nodes (8323s -> 1160s).

Thanks to Hawk we have been able to validate our algorithm on a new architecture with AMD EPYC processors and an InfiniBand HDR 200 Gbit/s network. Before this project, experiments were performed on the CINES Occigen supercomputer in France with Intel Haswell processors and an InfiniBand FDR 56 Gbit/s network. The results in terms of CPU times are rather good on this new architecture even if for some algorithms like the SVD, which can be hardly parallelised, using multithreading on the 128 cores of a node can be counterproductive and using less threads can give better results.

The developed approach now allows us to process large datasets faster and to start executions by coupling different samples taken at different times. The tool is now very functional for end-users.

2.12 Cut-off December 2020

2.12.1 Parallel high order finite element solver for the simulation of nanoscale light-matter interaction in disordered media, 2010PA5590

Overview:
DIOGENeS (DIscOntinuous GalErkin Nanoscale Solvers) is a software suite, which is dedicated to the numerical modelling of nanoscale wave-matter interactions in 3D. The initial (and current) version of this software concentrates on optical weave interactions with nanometer scale structures for applications to nanophotonics and nanoplasmonics. DIOGENeS relies on a two-layer object-oriented architecture. The core of the suite is a library of generic software components (data structures and algorithms) for the implementation of high order DG (Discontinuous Galerkin) and HDG (Hybridisable Discontinuous Galerkin) schemes formulated on unstructured tetrahedral and hybrid structured/unstructured (cubic/tetrahedral) meshes. This library is used to develop dedicated simulation tools for time-domain and frequency-domain problems relevant to nanophotonics and nanoplasmonics, considering various material models. The programming language adopted for all the components of DIOGENeS is Fortran 2003/2008. The present project aimed at improving the scalability of a DGTD (Discontinuous Galerkin Time-Domain) solver from the DIOGENeS software suite by implementing a fine grain task-based parallelisation using OpenMP, in addition to the already existing coarse grain distributed-memory parallelisation based on the MPI standard.

Scalability results:
The observed scaling is almost ideal up to 16 threads. Moreover, the scaling is better as the interpolation degree in the DGTD method is increased, which was an expected behaviour. Indeed, for a fixed mesh size, when the interpolation degree is increased from 1 to 4, the computation/data access ratio increases and is more favourable to a better scaling of the multithreaded parallelisation.

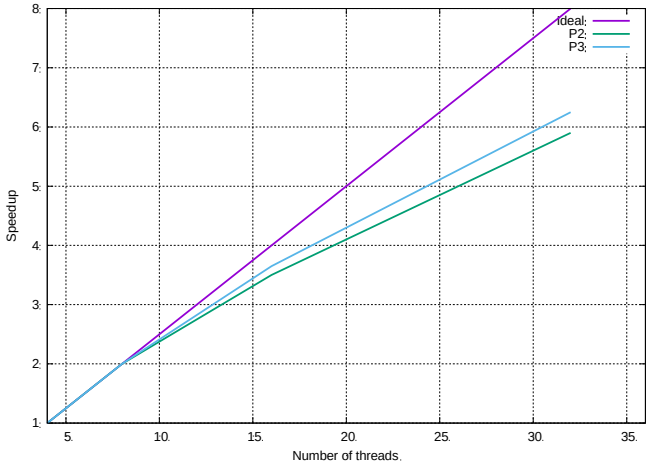


Figure 37: Strong scalability result achieved using 4 nodes, 4 MPI processes and a varying number of OpenMP threads for a fixed problem size

Accomplished work:
The main features of the DGTD solver are the following: (1) It is formulated on an unstructured tetrahedral mesh. In fact, the solver is even able to work with an orthogonal hexahedral mesh or a hybrid hexahedral-tetrahedral mesh, i.e. a hybrid structured-unstructured grid; (2) The main elementary numerical kernels are dense matrix-vector products, which are performed for many element-wise operations within loops over mesh elements (i.e. tetrahedra). The matrices involved in these operations are of relatively small size, up to a few hundred rows. In the current implementation of the DIOGENeS core library, we have hand optimised several matrix-vector product routines each for a particular value of the number of rows. A coarse grain parallelisation has been considered so far, which is based on a classical SPMD strategy combining a partitioning of the underlying mesh with a message-passing programming paradigm implemented with the MPI standard.

In summary, we successfully developed a fine grain parallelisation of the DGTD solver based on the OpenMP `TASKLOOP` pragma. The work that has been done in the project has consisted in the implementation of a fine grain parallelisation of the DGTD solver based on the OpenMP standard in order to improve the overall scalability of the solver on systems with SMP nodes. We decided to make use of the `TASKLOOP` pragma since this work is considered a first step before considering a data dependency graph-based task parallelisation of the DGTD solver, which will be conducted in a future study. These pragmas have been inserted in the compute-intensive routines of the DGTD solver, which all materialise as loops over mesh tetrahedra. Consequently, the `TASKLOOP` pragma is used to distribute these element-wise computations across threads. Each of these loops involve series of inner loops that correspond to dense matrix-vector products which are treated sequentially by each thread.

Main results:
The assigned PRACE system MARCONI100 has been used without difficulties. The available documentation on the web portal of the system has greatly facilitated the access, installation and compilation of the application code. Future work will be concerned with the further development of the task-based fine grain parallelisation by considering a data-dependency graph approach. This will allow to mitigate the computational load-balancing issues for physical applications involving highly heterogeneous propagation media involving a large number of particles. Besides, thanks to the availability of the hybrid MPI-OpenMP version of the DGTD solver resulting from the present project, it will be possible to conduct concrete studies of random lasing in very large domains.

2.12.2 PRACE QBee - Towards parallel quantum circuits for now, 2010PA5610

Intermediate status report:
In the beginning of this project, most of the work was focused on understanding the challenges in simulating a quantum computer and the main mechanics behind the most common method to simulate qubits, namely tensor mathematics. Then we analysed available open-source matrix-based simulators to observe its capacity and limitations, mainly memory usage, and the new state-based method of simulating qubits, using arithmetic operations that reduced exponentially the memory usage compared to the matrix-based simulation. Due to the need to distribute and parallelise computations, we have decided to design and develop a new simulator for quantum computing. The next step involved experiments with the new state-based method of simulating qubits and designing ways of distributing the state representation among different machines and processes using MPI. The first experiments were successful, we used the JUWELS Cluster

module at JSC to simulate qubits with a variable number of computers and processes. Using its large available system memory, we were also able to test the limits of our implementation and find the current limitations that impedes us to simulate a larger number of qubits. The current state of our work is in overcoming those limitations and testing our solutions using the JUWELS Cluster module.

Currently, the main obstacle is in tracking the position of the distributed quantum states across the various machines executing the simulator. The values to address such positions are exceeding the maximum value representable in an integer in C++ (uint64_t) when simulating a larger number of qubits. This limitation can only be found when simulating 34 qubits which uses 500 GB to 1 TB of total system memory.

2.13 Cut-off March 2021

2.13.1 Towards prototypical Rayleigh numbers in molten pool convection, 2010PA5685

Overview:
Nek5000 [16] is an open-source code for the simulation of incompressible flows. Nek5000 is widely used in a broad range of applications, including the study of thermal hydraulics in nuclear reactor cores, the modelling of ocean currents and the simulation of combustion in mechanical engines.

The Nek5000 discretisation scheme is based on the spectral-element method. In this approach, the incompressible Navier-Stokes equations are discretised in space by using high-order, weighted residual techniques employing tensor-product polynomial bases. The resultant linear systems are computed using Conjugate Gradient iteration (CG) with convenient preconditioners.

Nek5000 supports two distinct algorithms Pn-Pn and Pn-Pn-2 for solving the incompressible Navier-Stokes Equations. In this project, we will focus on the Pn-Pn-2 algorithm: first discretise in time and then take the continuous divergence of momentum equation to obtain a Poisson equation for pressure with special boundary conditions. When high-order backward-difference schemes (BDFk) in time are used, the assembled matrix can be written as a discrete Helmholtz operator.

Exascale HPC architectures are increasingly prevalent in the Top500 list, with CPU based nodes enhanced by accelerators or co-processors optimised for floating-point calculations. We have previously presented case studies of partially porting to parallel GPU-accelerated systems for Nek5000.

Originally, our goal for the project was to perform larger benchmarking tests with higher polynomial order using the next generation of Nek5000, NekRS, on the JUWELS Booster GPUs system. The OCCA [17] library is used in the NekRS as the backend program for different types of devices (e.g. NVIDIA GPUs). However, the OCCA library does not run with the system “mpi-setting” on JUWELS Booster. The issue has not been identified by JUWELS staff and there is no work-around available for this issue until end of the project. As a result, we focused on the performance and optimisation works for the CPU version on the JUWELS cluster system.

Scalability results:

The scalability tests were performed for numerical simulations of a 3D molten pool case on JUWELS cluster system at JUELICH. The case consists of a mesh of 763904 elements with polynomial order of 7. The total number of grid-point is around 250M.

The strong scaling performance, measured in the execution time per step in shown in Figure 38. For the strong scalability tests a speed-up of 22.8 can be achieved for up to 7680 cores (160 compute nodes) using 480 cores as baseline. The performance becomes worse with increasing the number of MPI ranks to 11520 (240 compute nodes). We have already known from previous experiences in the Nek5000 code that going below 50-100 elements per MPI rank reduces performance. The scalability result is as expected. For the CFD simulation in this project we only consider strong scalability tests.

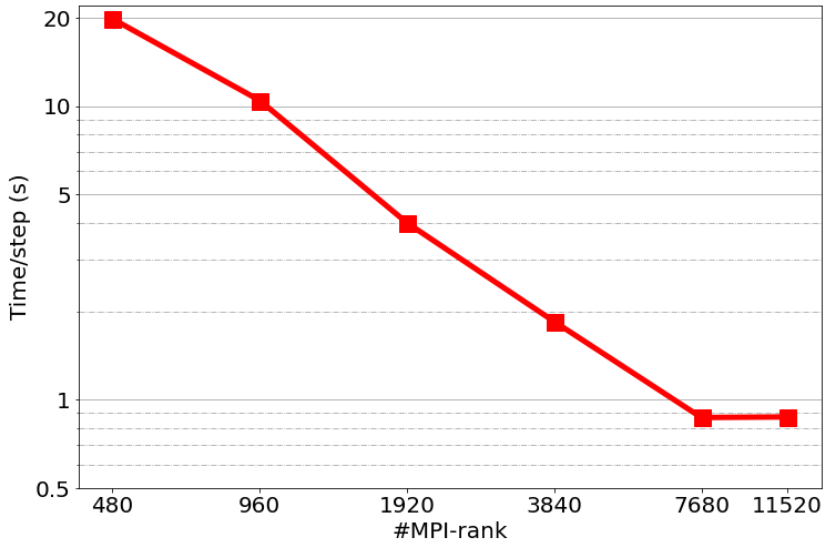


Figure 38: The execution time per step for the 3D case on JUWELS cluster system

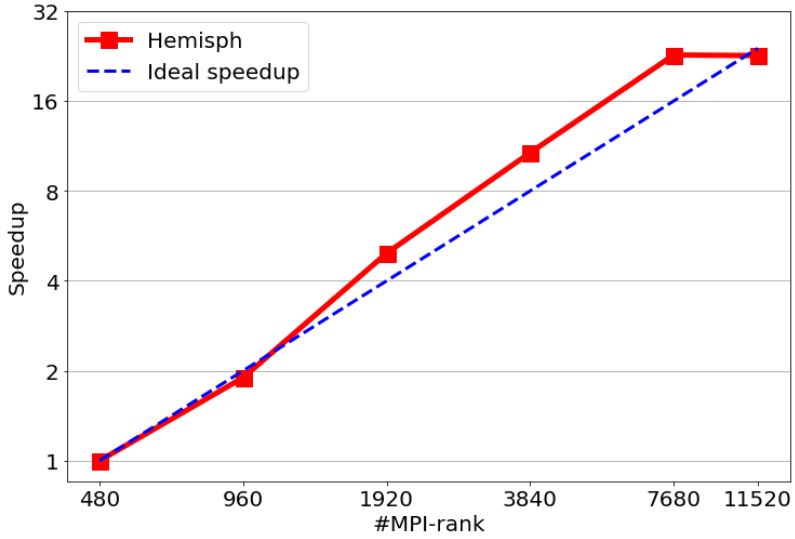


Figure 39: The speed-up for the 3D case on JUWELS cluster system

Accomplished work:

Actually there were run-time errors for this case with combining different compilers (i.e. GCC, INTEL, and NVHPC) and MPI libraries (i.e. OpenMPI, ParaStationMPI, and IntelMPI). Some combinations only work with a very specific number of MPI ranks. We spent much time to perform benchmarking tests using different compilers with MPI libraries. Finally, Intel compiler with ParaStationMPI is the only choice to obtain the strong scalability results between 480 and 11520 cores. The case run out of memory if less than 20 compute nodes (480 cores) were used.

By using the method introduced in [21], we investigated the linear interprocessor communication model

$$T = (\alpha+\beta m)T_a$$

where T is the communication time, m is the message length and T_a is the inverse of the observed Flop rate. The measured parameters on JUWLES system is,

$$\alpha = 3.8368e-06$$

$$\beta = 1.26138e-09$$

$$T_a = 1.425e-4$$

and Ping-pong test using 512 MPI ranks on JUWELS is shown below.

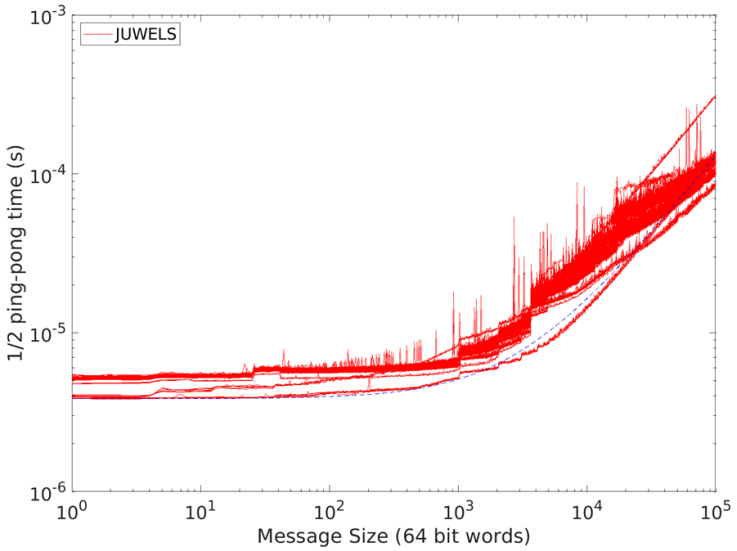


Figure 40: Ping-pong test using 512 MPI-rank on JUWELS cluster

Within this project, we investigated the performance of the XXT coarse grid solver for pressure. As an example, the profiling of the XXT coarse gird solver is shown below in Figure 41. The two functions apply_X2 and apply_Xt2, which are corresponding to be the sparse triangular solver, dominate the execution time. The two functions are not parallelised and have room for improvement.

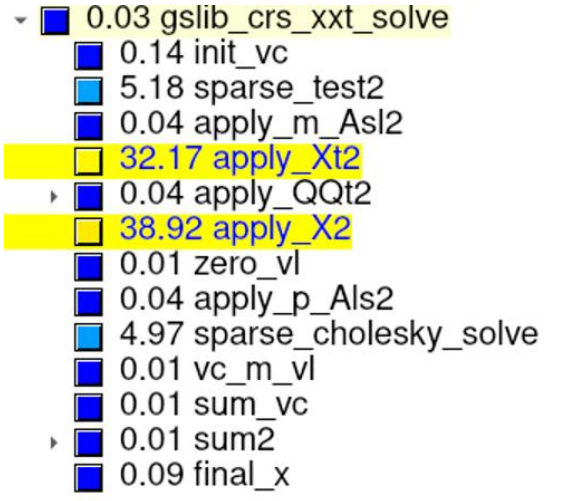


Figure 41: Profiling of the XXT coarse gird solver.

Main results:

We did not work on the highly customised version of software in production that required many changes for performance purposes. Instead, we focused on the production version of Nek5000 for the benchmarking tests with various MPI optimisation flags, which will be directly used for the larger simulation in the future.

We also investigated the communication models on the system and obtained corresponding parameters. That can be also used for the other cases on the system. A result of this project is that we have identified the XXT coarse solver for the pressure solver as a bottleneck in the CFD simulation that we need to improve. One approach to address the problem is to investigate alternatives such as HYPRE library for AMG preconditioning.

3 T7.2 Applications Enabling Services for Industry

In this section the progress and present status of the T7.2: Applications Enabling Services for Industry ("SHAPE") task will be described. Section 3.1 gives an overview of SHAPE along with some important statistics right up to the SHAPE 14 (most recent) call where the success in increasing the number of proposals received can be seen. The very successful SHAPE+ initiative is then described in Section 3.2 showing its impact in introducing new countries to SHAPE with 23 of the 26 eligible countries now having produced proposals.

The process for running SHAPE calls including the reviewing of proposals is as described in [22] and [4] and will not be repeated here.

The present status is described in Section 3.3. Projects from SHAPE 9-11 calls which completed around a year ago, and follow-up requests have been sent to the relevant SMEs and associated PRACE centres to see how the work performed with the assistance of SHAPE has affected the SMEs' business. These are reported in Section 3.4. Summary reports for the more recent SHAPE 12 and 13 call projects, and any on-going SHAPE 11 call projects, are reported in Section 3.5 along with an overview of the lessons learned. The SHAPE 14 call, which has recently closed, is described in Section 3.6.

3.1 SHAPE Overview

SHAPE (SME HPC Adoption Programme in Europe) is a pan-European initiative supported by the PRACE project within the WP7 work package. The programme aims to raise awareness and provide European SMEs with the expertise necessary to take advantage of the innovation possibilities created by High-Performance Computing (HPC), thus increasing their competitiveness. The programme allows SMEs to benefit from the expertise and knowledge developed within the top-class PRACE Research Infrastructure.

SHAPE is there to help overcome the barriers faced by SMEs and allow them to get a foot on the HPC ladder. Barriers often faced by SMEs can consist of a combination of a lack of in-house expertise, a lack of available staff effort, or little or no ready access to suitable hardware. Given there is no guarantee of a return on investment and that investment means committing time and money, an SME may consider that the risks are too great to try HPC on their own when prior experience is lacking. On the other hand, HPC has the potential to improve product quality via an enhanced performance and accuracy of their models, or by reducing time to delivery, or by providing innovative new services to their customers. Ultimately, these things can in turn increase the competitiveness of the SME. SHAPE is there to help in overcoming the barriers faced by SMEs and allow them to get a foot on the HPC ladder.

Successful applicants to the SHAPE programme get dedicated effort from a PRACE HPC expert as well as access to a suitable supercomputing resource, or other appropriate hardware, at a PRACE centre. In return the SME commits a comparable amount of effort and provides their domain expertise which will not automatically be available at a PRACE centre. In collaboration with the SME, the PRACE partner helps the SME try out their ideas for utilising HPC to enhance their business.

At the end of a SHAPE project the SME, in conjunction with the PRACE partner, creates a white paper. These can be viewed on the PRACE website [25] along with a number of Success Stories [26] that have been created based on successful past projects.

So far, SHAPE has awarded 74 SMEs effort across 13 calls and the 19 projects from the SHAPE 14 call for applications are under review at the time of writing.

Table 6 shows the calls, applications, approved projects and person months committed from PRACE so far in SHAPE and the chart in Figure 42 shows the countries of the SMEs who have applied to SHAPE and those who have been awarded projects.

Call	Call open	No. Proposals	No. Awarded	PMs
Pilot	Jun 13	14	10	35
2	Nov 2014 - Jan 2015	12	11	45.25
3	Nov 2015 - Jan 2016	8	8	30.75
4	Jun 2016 - Sep 2016	7	4	17
5	Mar 2017 - Jun 2017	8	6	20.75
6	Oct 2017 - Dec 2017	5	2	9.5
7	Apr 2018 - Jun 2018	6	3	16.5
8	Oct 2018 - Dec 2018	1	1	3
9	Apr 2019 - May 2019	7	5	20.5
10	Oct 2019 - Dec 2019	5	5	18.75
11	Apr 2020 - Jun 2020	10	7	28
12	Oct 2020 - Dec 2020	16	9	38.75
13	Apr 2021 - Jun 2021	5	3	14.5
14	Oct 2021 - Nov 2021	19	Under Review	Under Review
	Totals	123	74	298.25

Table 6: SHAPE proposals received and awarded by call. SHAPE 14 proposals are presently under review.

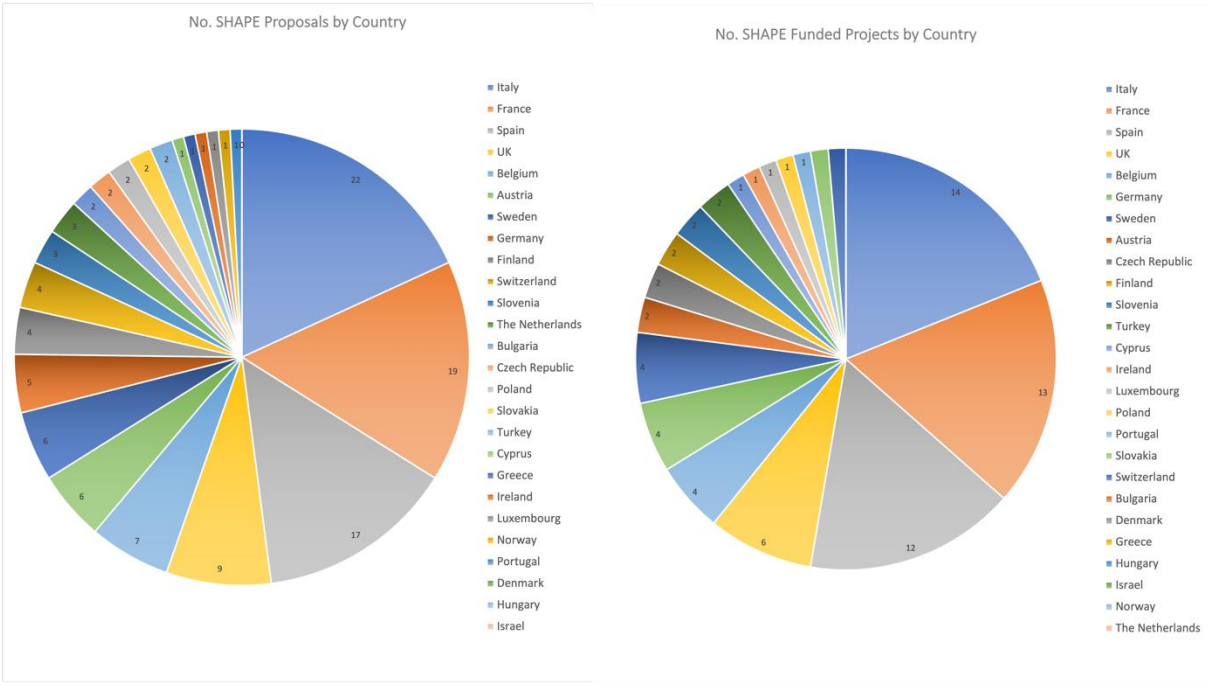


Figure 42: SHAPE proposals received and awarded by country

3.2 Increasing the number of participating countries: SHAPE+

Over recent years there have been a number of discussions within the IP project reviews and within the PRACE IAC about ways to increase the number of countries participating in SHAPE, as well as increasing the number of proposals overall. Perhaps not surprisingly, proposals tended to originate more frequently from SMEs within countries with centres that had the most effort available to work on SHAPE projects.

At IAC in November 2018 the SHAPE+ initiative was born to try to overcome the above.

The SHAPE+ initiative makes available an allocation of person months to centres for projects received where

- the SMEs are from countries where no (or very few) SHAPE projects have been awarded so far;
- the centre in question does not presently have enough effort to carry out the SHAPE project.

Following on from the IAC meeting, at the management board meeting in May 2019 it was agreed that 30 person months (PMs) would be made available for this initiative. The SHAPE+ initiative was implemented from the SHAPE 10 call onwards.

The initiative has worked very well from the start. Call 10 received the first proposals from Slovenia and Turkey, both of which were awarded. Call 11 received the first proposals from Cyprus and Portugal and a further proposal from Turkey, all of which were awarded, and a further proposal from Slovenia (not awarded). Call 12 received the first proposals from Slovakia, a further proposal from Slovenia and Austria, all of which were awarded (a proposal had been received from Austria before, but this had not been awarded so Austria was considered to be a "new" country). In total this allocated 25.65 PMs of the 30 PMs available.

Call 13 was then opened in Spring 2021 for projects, some of which would run during the original PRACE-6IP timeframe, and some that would run during the extension (Jan – Jun 2022) and Call 14 was opened in Autumn 2021 for projects to run entirely during the extension period. To enable this, further PMs were made available in addition to the remaining 4.35 PMs from the initial SHAPE+ allocation and these could be used for any PRACE centre that did not have PMs remaining. Call 13 attracted two further proposals from Austria (one funded) and Call 14 attracted 2 further proposals from Austria, a second proposal from Slovakia, and the first ever proposal from Norway. At the time of writing, the proposals from Call 14 are under review.

It seems clear that the SHAPE+ initiative has been a success. Figure 43 shows the number of countries of origin of SMEs in terms of proposals received and projects awarded up to and including each call. The number of countries represented has increased over time and that increase has accelerated for the last few calls. As of Call 14, we have now received proposals from 23 of the 26 eligible countries, resulting in projects from 19 different countries. This leaves just SMEs from Denmark, Hungary, and Israel with no proposals and Bulgaria, Denmark, Greece, Hungary, Israel, Norway, and The Netherlands without projects.

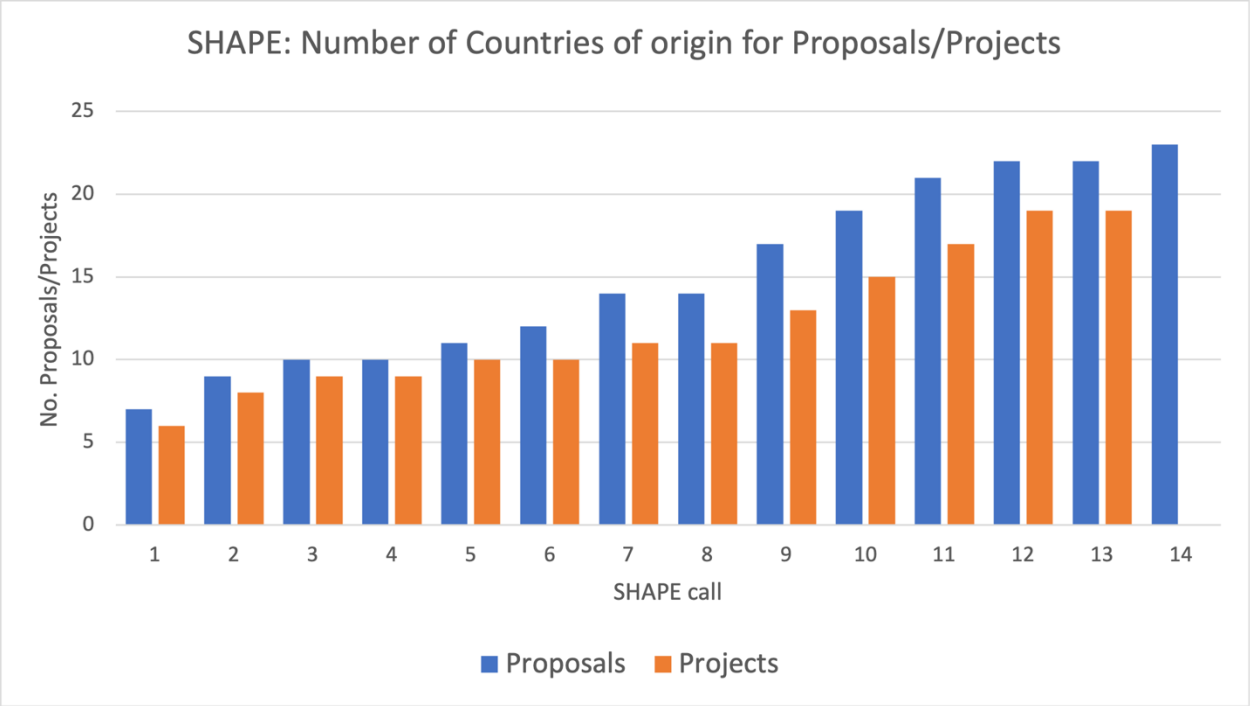


Figure 43: A graph of the increasing number of countries of SHAPE proposals and awarded projects. SHAPE 14 proposals are presently under review.

3.3 SHAPE Programme status

In this section, an overview of the status of all the SHAPE projects since the first (pilot) call are reported. SHAPE projects from the second call onwards fall into three categories:

1. Projects that finished 2 years ago or more which were fully reported in previous deliverables (e.g. PRACE-5IP D7.2 [22]).
2. Projects that finished around a year ago (SHAPE 9-11). Both the SMEs and PRACE partners from these projects have each filled in a survey to assess the impact of the SHAPE work on their business and the results are reported on in Section 3.4.
3. Projects that are ongoing or in the process of finishing (SHAPE 12-13 and 2 projects from SHAPE 11). PRACE partners from these projects have each provided a summary of their project. These are reported on in Section 3.5.

As of November 2021, the status of all of the projects is shown in Table 7.

Call	SME	PRACE Partner	White Paper	Report
2	WB-Sails	CSC	Published	See [23] [25]
2	Principia	CINES	Internal technical report	Not Provided, see [23]
2	Algo'tech	INRIA	Published	Not Provided, see [23] [25]
3	ACOBIOM	CINES	Published	See [23] [25]
3	Airinnova AB	KTH	Published	See [23] [25]
3	Creo Dynamics AB	KTH	Published	See [23] [25]
3	AmpliSIM	IDRIS	Published	See [23] [25]

Call	SME	PRACE Partner	White Paper	Report
3	ANEMOS SRL	CINECA	Published	See [23] [25]
3	BAC Engineering Consultancy Group	BSC	Published	See [23] [25]
3	FDD Engitec SL	BSC	Published	See [23] [25]
3	Pharmacelera	RISC	Published	See [23] [25]
4	Artelnics	BSC	Published	See [22] [25]
4	Milano Multiphysics	CINECA	Published	See [22] [25]
4	Renuda UK Ltd	EPCC	Published	See [22] [25]
4	Scienomics	IDRIS	Published	See [22] [25]
5	Disior Ltd	CSC	White Paper not expected due to change in focus of SME	See [22]
5	Invent Medical Group, s.r.o.	IT4I	Published	See [22] [25]
5	AxesSim	CINES	Published	See [22] [25]
5	E&M Combustion S.L.	BSC	Published	See [22] [25]
5	Svenska Flygtekniska Institutet AB	KTH	White Paper not expected due to SME unavailability	See [22]
5	Shiloh Industries Italia s.r.l	CINECA	Published	See [22] [25]
6	Axyon AI SRL	CINECA	Published	See [4] [25]
6	Vision-e S.r.l.	CINECA	Published	See [4] [25]
7	Briggs Automotive Company Ltd	STFC	White Paper awaiting final approval	See [4]
7	Polyhedra Tech SL.	BSC	Published	See [4] [25]
7	FLUIDDA	BSC	Published	See [4] [25]
8	Energy Way Srl	CINECA	White Paper not expected due to change in focus of SME	See [4]
9	Neuralbit Technologies	PSNC	Awaiting White Paper	See Section 3.4
9	NIER Ingegneria S.p.A.	CINECA	Published	See [25] and Section 3.4
9	LVD Biotech, S.L.	BSC	Published	See [25] and Section 3.4
9	Submer Immersion Cooling	BSC	Published	See [25] and Section 3.4
9	SPARC Industries SARL	U Luxembourg	Awaiting White Paper	See Section 3.4
10	OPEN ENGINEERING	IDRIS	Published	See [25] and Section 3.4
10	Global Surface Intelligence	EPCC	Published	See [25] and Section 3.4
10	Enlighting Technologies SL	BSC	Published	See [25] and Section 3.4
10	SPARK inovacije, d.o.o.	UL	Awaiting White Paper	See Section 3.4

Call	SME	PRACE Partner	White Paper	Report
10	Tarentum	UHEM	Awaiting White Paper	See Section 3.4
11	VIPUN Medical N.V.	UANTWERPEN	Internal Technical Report	See Section 3.4
11	Mobildev	UHEM	Published	See [25] and Section 3.4
11	Global Maritime Services Ltd (G.M.S.)	EPCC	White Paper not expected due to SME unavailability	Project Cancelled
11	Offshore Monitoring Ltd. (OSM)	CaSToRC	Awaiting White Paper	See Section 3.4
11	d:AI:mond GmbH	HLRS	Technical work ongoing	See Section 3.5
11	CENTIMFE	UC-LCA	Published	See [25] and Section 3.4
11	SmartCloudFarming GmbH	HLRS	Technical work ongoing	See Section 3.5
12	Integrative Biocomputing – IBC	IDRIS	Technical work ongoing	See Section 3.5
12	Queen Cassiopeia	Not assigned	Technical work ongoing	Project Cancelled
12	AIE	STFC	Technical work ongoing	See Section 3.5
12	akustone s.r.o.	IT4I	Technical work ongoing	See Section 3.5
12	TAILSIT	TU Wien	Technical work ongoing	See Section 3.5
12	3Tav d.o.o.	UL	Technical work ongoing	See Section 3.5
12	Voxo	ENCCS	Technical work ongoing	See Section 3.5
12	MultiplexDX, s.r.o.	CCSAS	Technical work ongoing	See Section 3.5
12	BuildWind SPRL	CINES	Technical work ongoing	See Section 3.5
13	Reintrieb GmbH	TU Wien	Technical work ongoing	See Section 3.5
13	SIRIS Academic S.L.	BSC	Technical work ongoing	See Section 3.5
13	Ingénierie et Systèmes Avancés	IDRIS	Technical work ongoing	See Section 3.5

Table 7: Complete list of SHAPE projects awarded to date

3.4 SHAPE 9-11: Follow up for completed projects

This section focuses on SHAPE projects completed in 2019-2020. SMEs involved in projects that completed in this time frame will in most cases have now had time to assess the impact of HPC on their business. For the last two deliverables [22][4] in consultation with the PRACE

IAC, two on-line surveys were constructed, one for the PRACE partner to fill in, the other for the SME. The PRACE partner survey gathers information about the technical improvements arising from the project while the SME survey gathers information about the business improvements. The template for the surveys were given in the appendix of [22] and, as it was considered a great success, other than minor improvements it has not changed since then. These surveys provided useful quantitative and qualitative information and so have been repeated for this deliverable.

In total 13 relevant PRACE partners and SMEs were surveyed. 11 of the partners and 9 of the 13 SMEs responded. The full responses are presented below in sections 3.4.1 to 3.4.6 and contain information gathered in both the PRACE partner survey and the SME survey.

Below are the highlights of the results from surveys to SMEs and partners together with the questions.

3.4.1 HPC before and after SHAPE

SMEs were asked a set of questions based on their use of HPC before and after the SHAPE project, and the number of employees with an HPC background:

Were you using HPC before SHAPE?

With regard to your current HPC usage, are you

- *Not using HPC at all?*
- *Using Cloud-based HPC Services?*
- *Using your own HPC Systems?*
- *Using resources from a PRACE HPC centre?*

Do you have plans to do more with HPC in the future? (If yes, please give details)

How many employees had an HPC background before the SHAPE project?

How many employees have an HPC background now?

The results are shown in Table 8.

SME Name	Using HPC before SHAPE?	Using HPC after SHAPE	No. Employees with HPC skills before SHAPE	No. Employees with HPC skills after SHAPE	Rise in employees with HPC skills	Do you have plans to do more with HPC in the future?
Submer Technologies	No	Yes: Using Cloud-based HPC Services	1	2	1	Yes: Thermal simulations
NIER Ingegneria	No	Yes: There is a chance we will investigate use of cloud services	0	1	1	Yes: AI and ML algorithms and data analysis
Global Surface Intelligence	Yes	Yes: Mix of PRACE HPC and cloud	3	2	-1	Yes: Cloud migration and scaleup.

SME Name	Using HPC before SHAPE?	Using HPC after SHAPE	No. Employees with HPC skills before SHAPE	No. Employees with HPC skills after SHAPE	Rise in employees with HPC skills	Do you have plans to do more with HPC in the future?
Enlighting Technologies	No	No: Not using HPC at all	0	1	1	No
Offshore Monitoring Ltd.	No	Yes: Using resources from a PRACE HPC centre	1	2	1	Yes: We plan to analyze and implement further how HPC frameworks can help us in analysis and processing of high volumes of metocean geospatial data, and also to improve the performance of our multi-objective route shipping route optimisation service.
Mobildev Kurumsal Hizmetler İletişim A.Ş.	Yes	Yes: Using Cloud-based HPC Services	2	2	0	Yes: We intend to use it again for model training and analysis in the projects we are developing.
SPARK inovacije	No	Yes: Using resources from a PRACE HPC centre	0	3	3	Yes: We will probably have many complex optimisation problems, which we will need to solve on a daily basis. We'll try to do a further enhancement of the current code, to include distributed memory paralelisation.
TARENTUM	Yes	Yes: Using Cloud-based HPC Services	3	4	1	Yes: Several weather simulation results are needed for our products.

SME Name	Using HPC before SHAPE?	Using HPC after SHAPE	No. Employees with HPC skills before SHAPE	No. Employees with HPC skills after SHAPE	Rise in employees with HPC skills	Do you have plans to do more with HPC in the future?
CENTIMFE	No	Yes: Collaboration with LCA-UC	0	1	1	Yes: Collaboration with University of Coimbra for developing services for SMEs and for research.
	Yes = 3; No = 6	**Yes = 8; No = 1**	**Average = 1.1**	**Average = 2**	**Average = 0.9**	**Yes = 3; No = 6**

Table 8: SME HPC usage and experience before and after a SHAPE project

As can be seen, there has been a significant uptake in the use of HPC (from 3 to 8) and almost a doubling of the number of employees with HPC skills. The majority are using cloud-based services and/or PRACE HPC resources.

SMEs were also asked

Will you seek HPC expertise from elsewhere?

Three reported "yes" adding

- Outsourcing
- Cloud migration - not strictly HPC but associated - is supported by external consultancy. Their role is strictly design & support, rather than implementation.
- We are included in the POP CoE project.

3.4.2 Return on Investment (RoI)

SMEs were asked

Has your project led to a measurable return on investment, or a predicted return in the coming months?

Three of the SMEs reported "yes" here, of these, one reported that the return was not measurable at this stage, one giving a 5% increase, and another further explaining: "Our route optimisation service is at around TRL6 / TRL7 and not yet commercialised, so we do not have revenue at this time from the service, but it is estimated that the increase in performance when HPC frameworks are implemented, will help us in acquiring 10-20 % further customers and similar revenue increase as per our projections currently.".

3.4.3 Business processes

SMEs were asked

How has the SHAPE project affected your business process?

- *Cost reduction,*

- *Faster time to market,*
- *More sales,*
- *Improved R&D process*
- *Improved service to customers*
- *Not at all,*
- *Other."*

From Table 9 we can see that the majority of SMEs (7) have reported improvement in the research and development methodologies with several (4) reporting improved services to customers. Others report cost reductions and a faster time to market. Only one SME reports no change to their business processes.

	Cost reduction	**Faster time to market**	**Improved R&D process**	**Improved service to customers**	**Not at all**
Submer Technologies			✓		
NIER Ingegneria			✓		
Global Surface Intelligence	✓			✓	
Enlighting Technologies			✓	✓	
Offshore Monitoring Ltd.			✓	✓	
Mobildev Kurumsal Hizmetler İletişim A.Ş.	✓		✓		
SPARK inovacije		✓	✓	✓	
TARENTUM		✓	✓		
CENTIMFE					✓
Total	**2**	**2**	**7**	**4**	**1**

Table 9: SMEs changes to their business processes due to a SHAPE project

3.4.4 Business outcomes

SMEs were asked:

What have been the main business outcomes of your project?

SMEs replied:

- Submer Technologies: Increased knowledge on our technology.
- NIER Ingegneria: Know how for new services.
- Global Surface Intelligence: PRACE project was essential element to ongoing project to scale up & automate production, and cloud migration. Main benefits have yet to be realised as larger project is ongoing. Expected outcomes of larger project: 1) Lower costs, 2) faster deliver, 3) increased margins, 4) growth in turn over and market share.

- Enlighting Technologies: The project was focused on the light delivered by luminaries installed in a hospital. We were interested in HPC for having a better resolution of the results.
- Offshore Monitoring Ltd: Improvement in performance of the shipping route optimiser service.
- Mobildev: Thanks to the model developed after the study, the planned work was solved both faster and more cost-effectively.
- SPARK inovacije: We have enhanced our algorithm and code, which can provide good results faster.
- TARENTUM: It helped us with better prediction.
- CENTIMFE: Demonstration of competences. Research projects.

3.4.5 Value of SHAPE

SMEs were asked

Fundamentally, has participation in SHAPE been of real value to your SME?

Every SME reported "yes" to this question, with further explanation given as:

- Submer Technologies: Increased knowledge on our technology.
- NIER Ingegneria: Acquisition of know how.
- Global Surface Intelligence: Critical element to larger project.
- Enlighting Technologies: We had access to a very powerful tool to have better results, which is related to more precision in the study that will be published.
- Offshore Monitoring Ltd.: We have analysed and evaluated how HPC utilisation can improve the performance of optimisation functions and solutions, and we will utilise this knowledge in our continuous upgradation of the system.
- Mobildev Kurumsal Hizmetler İletişim A.Ş.: Thanks to the academic support and HPC we received, our work was completed much faster and more stable.
- SPARK inovacije: Our improved algorithm achieves good results faster, which in turn provides more value (savings in logistics costs, emissions, etc.) to our customers.
- TARENTUM: Since it give us better prediction which directly related to precision our products.
- CENTIMFE: Faster numerical simulations.

The SME was asked

Would you recommend SHAPE to other SMEs?

All reported "yes".

Are there any other benefits that have arisen from the SHAPE project, and if so please describe?

- Submer Technologies: No.
- NIER Ingegneria: networking and very positive collaboration with the CINECA staff.
- Global Surface Intelligence - Closer connection to PRACE partners. Deeper understanding of HPC within the company.
- Enlighting Technologies: Know-how for the employees of the SME.

- Offshore Monitoring Ltd.: Contact with a PRACE centre, and direct link with them for further work and collaborations together. Knowledge of availability of various HPC centers in Europe.
- SPARK inovacije: We have stronger ties with HPC centre and community, we have also published an article at the SOR conference, which contributes to our recognition as being scientifically excellent in the business community.
- TARENTUM: We now know the HPC resources of Europe through SHAPE experience.
- CENTIMFE: New possibilities for research projects.

The PRACE partner was asked

Has the project led to new or enhanced software features? If yes, please give details

- SUBMER: Yes, we have implemented an anisotropic porosity model to account for CPUs/GPUs heat sinks. In terms of methodology, we have calibrated these models using individual fully resolved heat sinks. These models enabled us to solve the full immersion cooling setup developed by the company.
- Enlighting Technologies SL: Yes, we developed a new version of the code allowing us to run it on multinode using MPI. Also, we enhanced the input and output management. Additionally, we parallelised a serial part of the code taking most of the time. With these changes, the SME is able to run the software faster and with larger resolutions.
- VIPUN: Yes, the work provided new insight in the data and algorithms that are to be implemented in the product they are developing.
- Sparc Industries: Yes, performance improvements (x100)
- Open Engineering: Yes, a multi-GPU implementation of the code has been performed and was mostly based on asynchronous kernel launches on each available GPU device on the node to reduce latencies.
- Offshore Monitoring: No
- NIER Spa: No
- MOBILDEV: Yes, the project allowed to conduct training on large data set and add more features to the software.
- SPARK Inovacije d.o.o.: Yes, the code that they have developed before the PRACE SHAPE project, was during the project enhanced:
    - parallel multi-seed computation: We have run several threads, each one with its own, different seed. This random seed impacts the rate with which a better solution is found. It is impossible to anticipate which random seed will give us better results for which problem. Using several parallel processes with different seeds we have increased our chance to find a good solution within a given time.
    - Step method: We have tried to take advantage of the fact that some seeds were more “efficient” than the others, their convergence was faster. We used the currently best solution as an initial solution for the rest of the process. The process goes as follows:
        - Run the algorithm with multiple seeds as above in parallel.
        - Stop each branch of the algorithm after a certain number of iterations and find the best (cheapest) solution found so far among all the random seeds.
        - Select this solution, set it as the initial solution (starting point) for all the random seeds, and continue with the next step.
        - Repeat until the termination criterion of the main process is met.

    - The idea for this method arose from existing approaches:
        - Gradient descent - we are using the best solution as a start for the next step. The difference is that we are not able to analytically determine the best "direction" for continuation, but have to heuristically search the neighbourhood, using several threads with different seeds.
        - Genetic algorithms - a single step produces a "population", and the best representatives of the population are used for the next step. The difference is that we have used only one best solution for the next step. We could potentially also use all non-dominated solutions, or something similar, but would have to determine which new thread gets which initial solution. This might be feasible, but would need some additional mental effort.
- TARENTUM: No
- GSI: Yes, a new software package, Moonshine, was developed by EPCC as proof-of-concept for a bespoke workflow management framework to better handle GSI's multiple complex data workflows, including data fetching, data processing, data analysis and machine learning. Moonshine is part of core functionality for a new workflow pipeline being developed by GSI that will exploit HPC resources as well as cloud compute.

Partners were asked

What has been the technical impact of your project?

The results are show in Table 10.

	Ability of a simulation (previously serial) to run on HPC systems	**Improved scalability of code**	**Ability to handle increased number of data points**	**Improved workflow**	**Increase speed of software package**	**Parameter estimation of algorithms using optimisation techniques using HPC resources**	**Improved resolution of simulations**
SUBMER				Y			
Enlighting Technologies SL	Y	Y	Y		Y		
VIPUN				Y		Y	
Sparc Industries	Y	Y	Y				
Open Engineering		Y			Y		Y
Offshore Monitoring					Y		
NIER Spa	Y			Y	Y		
MOBILDEV			Y		Y		
SPARK Inovacije d.o.o.	Y	Y	Y		Y		
TARENTUM							
Total	**4**	**4**	**4**	**3**	**6**	**1**	**1**

Table 10: Technical impacts of a SHAPE project

Partners were asked

Please describe any performance improvements made to the SME's code.

Replies were given as follows:

- SUBMER: The code used is the available source code developed at BSC, Alya.
- Enlighting Technologies SL: Parallelisation of a serial part of the code obtaining 3.25X of SpeedUp on single node. With the enhanced input/output management we obtained more than 25X of speedup for a 100 nodes job.
- VIPUN: improved visualisation of results, alternative algorithms for data analysis.
- Sparc Industries: x100 on prototype.
- Open Engineering: Improvements in speedup and parallel efficiency have been observed for every tested system. The Open Engineering RF solver is now compatible with modern HPC infrastructure. With the work done in this project, the code can now benefit from the speedup provided by several GPUs on a single compute node. That was the major goal of the project and it has been achieved.
- Offshore Monitoring: GPU code for the wave resistance calculation.
- NIER Spa: Pre-processing video: 8X on 32 cores, NN parallel training: 1.88X on 4 GPU (the dataset was very small).
- MOBILDEV: With the help of experience researchers, the code improvement on memory usage and time complexity.
- SPARK Inovacije d.o.o.: It is described above. The main point is parallelisation of space search, where each trajectory started with a different random seed. This task is almost linearly parallelisable: the number of points we can evaluate increases linearly with the number of processes.
- TARENTUM: They run the open software package WRF.
- GSI: Not applicable.

3.4.6 Summary of results

It seems clear from the results that SMEs value SHAPE with all SMEs reporting that their project provided them with a valuable experience. Clear improvements in business processes, improved research and development processes, and improved services to customers have been reported along with impressive speed up in code performance. SMEs have benefitted from a mixture of HPC experiences on CPUs and GPUs with a clear gain in the number of SME employees who understand the benefits of HPC.

3.5 On-going SHAPE 11-13 calls: Project summaries

Here we present the on-going SHAPE projects and those recently finished.

3.5.1 d:AI:mond (Germany)

Project Partners
SME contact: Thilo Krüge
PRACE contact: Anna Mack (HLRS)

Overview
As a services company, d:AI:mond GmbH supports its customers in a wide range of data science projects. The focus is on data-driven problem solving, where data gathered by customers is analysed and tailored solutions are developed. Major projects are predicting (chemical) product quality from sensor data, integration of data collected over many years from Excel sheets or prediction of daily turnover. Customers come from various industries, such as the chemical industry or the automotive industry.

The goal of the project is to optimise the production sequence in a company that produces general cargo. To do so, a detailed discrete event simulation of the complete production process will be implemented. Once this is done, this simulation will be used to create training data for a reinforcement learning model based on a neural network.

Activity performed
As a starting point in the project there was a first simple discrete event simulation in Python using the framework SimPy. The workspace and Python framework environment were set up on the HLRS system HAWK. The simulation was extended and improved. For the output generation the Pandas framework is used. A farming program was implemented in order to run many simulations in parallel using MPI. A jobscript was created to submit the farming program on HAWK using a single node running one simulation on every core. Randomised input data is generated in a separate step of pre-processing. Post-processing was implemented as parallel data gathering to a single output file using the mpi4py framework. The jobscript was updated to run simulation, pre- and post-processing as a whole pipeline. A neuronal network was implemented to generate the optimal input parameters for following simulations based on the output of prior simulations. The feedback of optimised data is currently in the making which includes also a file locking strategy and the termination of the loop depending on convergence of the result. The current parallelisation strategy intends to run 127 simulations in parallel and one optimisation step independently.

PRACE cooperation
In a weekly meeting, Anna Mack gave support in machine access, deployment of frameworks, parallelisation strategy, job submission and workspace management.

Benefits for SME
SHAPE is a very good opportunity for d:IA:mond in two ways. First, it can accelerate the pilot project that is already running with a local general cargo company. The simulation is computationally more costly than expected and has to run concurrently anyway. Second, the demanded white paper will be used for promotion purposes. Such a white paper is more valuable for the company, if it is written within the scope of a big, funded project like SHAPE.

Lessons learned
PRACE coordinator Anna Mack: Since this is my first SHAPE project to be responsible for and the topic also needed some research for me the timescale was quite tight. The communication was really good and the level of know how complemented each other well.

SME contact Thilo Krüger: The main issue of the current project is that the run time of our simulations is much higher than in earlier projects. Therefore we needed to find out if, and if yes how we can parallelise those simulations in a useful way. We learned the following lessons:

1. It is possible to parallelise the simulations and to optimise the underlying process by using the results of the collected simulations.

2. In the current use case more than 128 parallel simulations are unnecessary since in 128 simulations, we collect more than enough data for one round of the optimisation part.

As expected, it is not necessary to do the optimisation part on GPUs, since the simulation part is much slower and the optimisation part can be done on one of the cores.

3.5.2 SmartCloudFarming GmbH (Germany)

Project Partners
SME contact: Michele Bandecchi
PRACE contact: Anna Mack, Sameed Hayat (HLRS)

Overview
This SME analyses soil data collected via remote sensing. This can be integrated into smart farm machinery. They are developing a machine learning model deploying deep learning models on large amounts of satellite data. They wish to try this out on PRACE hardware and refine their models.

Activity performed
The first step for the project was to set up the environment and get the code running with GPU support. This was completed successfully. The code provided by PRACE required the files to be downloaded from the internet and after download running the model but due to the limitation of the system, as internet access is restricted, the data had to be pre-downloaded. This required the decoupling of the download logic from the processing logic, as the code provided by PRACE initially had all the logic merged. It was agreed that a separate script will be provided for download and processing so the data can be pre-downloaded and then processed.

PRACE cooperation
The project is still on going and HLRS and the SmartCloudFarming team are working together on the project.

Benefits for SME
To be completed beyond the end of the project.

Lessons learned
To be completed beyond the end of the project.

3.5.3 Integrative Biocomputing – IBC (France)

Project Partners
SME contact: Pridi Siregar
PRACE contact: Camille Parisel, Thibaut Very, Isabelle Dupays (IDRIS)

Overview
The SME specialises in Computational biology and has Machine Learning (ML) and Deep Learning (DL) programmes written in Python and knowledge-base systems (KBS) written in C++, with I/O based in Excel. The aim of the project is to try out HPC using GPUs with AI/ML drug design.

Activity performed
The project has not started yet.

3.5.4 AIE (UK)

Project Partners
SME contact: Nathan Bailey, Shahidur Mohammed,
PRACE contact: Daniel Ward, Andrew Sunderland (Hartree Centre, STFC)

Overview
Advanced Innovative Engineering (AIE) is an engineering company specialising in the development of innovative rotary engines for unmanned vehicles. From its headquarters in Lichfield, United Kingdom, it manages entire project lifecycles through concept, prototype and production. Working with international partners and customers, AIE creates technologies that combine low total-cost-of-ownership with exceptional reliability and versatility for global commercial and defence markets.

AIE are currently developing Wankel engine technology. These are rotary combustion engines with a high power to weight ratio, this feature making them particularly appropriate for applications in drones and other Vertical Take Off and Landing (VTOL) vehicles, either as primary power or as a range extender/backup for an electric power unit. One of the limitations of Wankel engines is they typically need active cooling. AIE's advancement in their SPARCS technology is to use pressurised air cooling, thus saving weight. It is understood that this technology is working on some of the larger engines, but heat rejection is harder on the smaller engines which are particularly demanded by the market.

They have sought out the Hartree Centre via the PRACE SHAPE project to help with the fluid dynamics, such that they might be enabled to run larger scale simulations on our resources. While AIE have engineers who are skilled using CFD packages on desktop applications, the intention in this project would be to get them using HPC to provide them with the fidelity to be able to optimise the heat exchanger design.

Hartree is helping AIE use HPC to gain the capability to do high fidelity modelling and simulation to optimise air cooling heat exchanger designs. The goals of the project are to provide training in meshing for large scale OpenFOAM jobs and set up of simulation environment for HPC calculations, to look at suitable workflow optimisations with AIE and to look at visualisation workflow, providing training and capabilities to extract metrics as defined for the project by AIE. A final aim is to identify suitable platforms for AIE to continue HPC-based research independently, if required, beyond the end of the SHAPE project.

Activity performed
- To date the main project work has involved liaising with AIE to generate a computational mesh. This has involved setting up an OpenFOAM-based computation including: mesh generation on a variety of geometries, determining appropriate boundary conditions, and determining convergence criteria.
- Running mesh independence study on Scafell Pike (ongoing).
- Training with AIE to prepare them to use the OpenFOAM case.
- Developing a post-processing workflow.

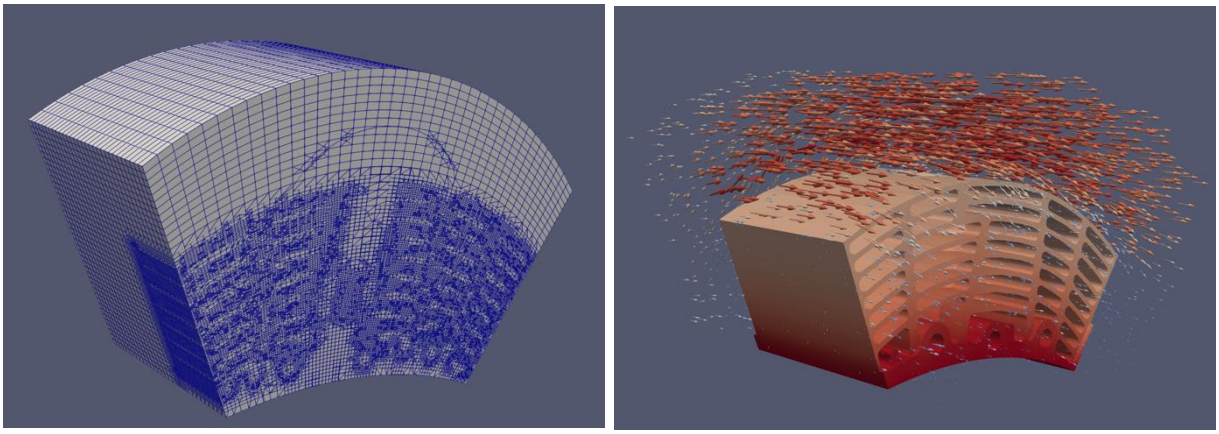


Figure 44: (Left) Example of mesh, (right) example of simulation with contours of temperature and arrows representing flow velocity.

PRACE cooperation

Aside from the involvement of STFC in the project, PRACE has been involved in guidance for developing the original project proposal, advising on scope and HPC elements of the project. Both AIE and STFC staff are made aware of any relevant PRACE training available (e.g. OpenFOAM) throughout the course of the project.

Benefits for SME

Due to the large number of design and parameter iterations required to achieve an optimised heat exchanger design, sheer computing power becomes a significant factor in reaching a prototype design within reasonable timescales. Given this, the use of HPC will have a substantial impact on the overall success of the new system.

Through high fidelity simulation using HPC, we hope to achieve a 2-3% cooling improvement over our baseline design. This improvement would be critical to addressing the targeted markets (ultra-lightweight hybrid UAV's) for this new power unit; these markets could be worth up to £6m per annum to AIE once the technology is proven and established.

Lessons learned

To be completed beyond the end of the project.

3.5.5 akustone s.r.o. (Czech Republic)

Project Partners

SME contact: Jakub Fojtík
PRACE contact: Tomas Karasek, Tomas Brzobohaty (IT4I)

Overview

Akustone s.r.o. is a small company focused on the development and production of stoves with a high heat storage capacity that serves as the comfortable radiant heaters. We are the only manufacturer of this type of stove in the Czech Republic. Stoves are made of heat-resistant natural material – soapstone. This natural mineral has 7 times higher heat storage capabilities than commonly used fireclay.

The main goal of this project is to replace the conventional methodology of flue gas path design of the soapstone stove with virtual prototyping using numerical modelling and simulation technologies. Because Akustone has no prior experience with numerical modelling, the implementation of virtual prototyping techniques into their design cycle will be done in several

stages. Methodology for virtual prototyping of this type of product will be created based on numerical simulation employing HPC technology. Full multiphysics CFD simulations for small time scale in combination with finite element simulation for large time scale will be created. Comparison of the numerical simulation results with available data from measurements will be done. We would like to use the results of numerical simulations for improvement of the stoves storage capacity based on new design modifications.

Activity performed

At the beginning of the project, identification of one stove type for numerical simulation, together with a collection of all necessary data as a 3D CAD model, material parameters, and information on typical loading conditions was done. Methodology for creating a virtual model of the stove was established. This methodology consisted of several computational strategies. An automated meshing procedure for creating a 3D numerical model based on CFD was created by using technologies integrated in OpenFOAM. The basic model of the isothermal flow in stove flue gas paths was created for a better understanding of the flow behaviour in these paths. This model was created in commercial tool CFX and also in open-source OpenFOAM.

Full multiphysics CFD simulation in combination with fluid flow and heat transfer in the stove was followed. We performed time-dependent CFD simulation in the full stove model with predefined boundary conditions that met the criteria set out in the experimental measurements. A complex mixture of gasses released from wood during the process of combustion was substituted by methane because combustion of methane is simple to model and there exists a global chemical reaction and methane is one of the major species of volatile matter released from biomass during the process of pyrolysis. The global chemical reaction of methane combustion is:

$$CH_4 + 2O_2 + N_2 \rightarrow CO_2 + 2H_2O + N_2$$

The equivalent mass source of methane was calculated on the following presumptions: (i) Heat released by the wood must be the same as heat released by methane at the same time. It can be expressed as the power of the fireplace in Watts, (ii) Moisture in wood is fully evaporated. LHV of wood is lower by evaporating water, (iii) Methane is released on the surface of volume, which represents a piece of wood. This procedure allowed us to estimate the mass source of methane per surface of wood piece with respect to energy equivalence and heat power of fireplace respectively. Generally speaking, a roughly simplified combustion process was able to predict basic interaction of hot flue gas with walls of fireplace and relatively precise describe internal aerodynamics of combustion chamber and grate.

PRACE cooperation

In this project, SME received support from PRACE in terms of expertise in CFD simulations. The developed solution was explained to the company and guidance and coaching will be provided in the future when the company will implement numerical modelling and simulation into their design process.

Benefits for SME

Solution and methodology developed within the SHAPE project help us improve our designs and even create new designs of stoves with better efficiency. A better understanding of the effects of burning processes and flue gases will lead to the stove design with lower emissions which could be used for passive houses where heat sources with high energy efficiency are necessary to meet requirements imposed by the standard. We see several benefits of the proposed work. In the short term, we expect to improve our existing design of the stoves and

thus increase the quality of existing products. In the medium term, this solution will help us to reduce the time-to-market of new stoves designs developed based on the results of numerical simulations. With new products, we expect to increase the number of customers and thus we will have to hire new people to be able to meet demand. We also foresee that the new stove design should have an impact on the environment due to lower emissions of CO_2 and other pollutants.

In the near future, we would like to cooperate (SME and research institution) in the possibility of designing new products in the form of joint research projects.

Lessons learned
The SHAPE programme seems to be a good option for SMEs, especially for those enterprises that do not have know-how in HPC technologies. It is necessary to mention not only HPC technologies but for example the area of numerical simulation itself. From the point of view of the cooperating institutions, SMEs and research institutions, the SHAPE programme is one of the few options how to start transferred know-how in digital technologies.

3.5.6 TAILSIT (Austria)

Project Partners
SME contact: Jürgen Zechner (CEO), Thomas Rüberg
PRACE contact: Soner Steiner, Claudia Blaas-Schenner (VSC Research Center of TU Wien)

Overview
This SHAPE project deals with electromagnetic simulations with finite/boundary element method for large systems using HPC. TAILSIT is a company based in Styria, Austria. TAILSIT's trademarks are custom-fit simulation software tools for electromagnetic problems and structural analysis including a numerical software library based on a coupled Finite/Boundary Element Method (FEM/BEM) for the efficient analysis of electromagnetic and structural problems. A major building block of TAILSIT's software portfolio is the BEM method which shows quadratic complexity with respect to system size in terms of runtime and memory requirements. For this reason, TAILSIT also implemented the so-called Fast Multipole Method (FMM) where the problem complexity reduces to almost linear scaling. But even with this acceleration technique, the use of the BEM method is rather limited on average desktop workstations.

During this project the main goals for the SME are/were:

- Get access to the HPC resources of VSC-4, the current flagship of the VSC family.
- Get consulting from the HPC experts of the VSC Research Center of TU Wien, which ultimately leads/led to knowledge gain of the SME.
- Improve the development of their parallel tree code using the Message Passing Interface (MPI).
- Test the code on an HPC infrastructure (VSC-4).
- Investigate the scaling (strong and weak) and efficiency of the code on the HPC infrastructure.

Activity performed
- A careful performance analysis revealed several bottlenecks and limitations, which were addressed by implementing optimised communication strategies with MPI.

- While the previous version of TAILSIT's software simulation relying on shared-memory parallelism was restricted to typically less than one million surface degrees of freedom, the new optimised version employing distributed-memory parallelisation with MPI allows to treat problems up to 50 x 109 surface degrees of freedom.
- The access to the HPC resources allowed TAILSIT to further test and optimise the shared-memory optimisation.
- A very good strong scaling in the FMM method could be verified (details of the scaling analysis will be presented in the white paper).

PRACE cooperation

- PRACE provided the necessary person months for one of the HPC experts of the VSC Research Center of TU Wien, while the HPC resources of VSC-4 were provided directly by the VSC Research Center of TU Wien, the Austrian PRACE-6IP partner.
- All three employees of TAILSIT have access to the VSC-4 cluster and they have been and still are using it ambitiously.
- Weekly meetings helped to identify and resolve problems regarding compilation and test runs of TAILSIT's software.
- Coaching and guidance on the HPC cluster was provided on a regularly basis.
- Several times optimised communication strategies with MPI were explained in-depth to the SME.

Benefits for SME

- This SHAPE project allowed for a giant leap from running simulations on desktop workstations to an HPC version of the software being able to not only treating much larger problem sizes than previously possible but also providing much faster time to solution.
- Larger models become more and more relevant to TAILSIT's customers and being aware of their customer's interest in 'faster code', the new HPC version helps TAILSIT to increase the competitiveness of their software library and this in turn puts TAILSIT in a much better market position.

Lessons learned

During this project both sides definitely learned a lot:

- After a warm-up phase, the communication and the contribution from both sides was really fruitful, e.g. through testing and profiling several bottlenecks could be identified, which led to discussions, suggestions, and finally to improvement of the software.
- At the beginning the communication was not so smooth since both sides did not know what to expect. TAILSIT was not sure which parts of the code they wanted to show to the HPC experts and it took a while to build mutual trust and allow for openness in sharing all parts of the code.

3.5.7 3Tav d.o.o. (Slovenia)

Project Partners

SME contact: Tomaž Čegovnik, Dejan Meštrić
PRACE Contact: Janez Povh, Pavel Tomšič (UL)

Overview

- Brief description of the SME

    - 3TAV is an established Slovenian company in the area of information solutions development. It was founded in 2001 and provides solutions in the area of information technologies for comprehensive support to all business processes in a company. Their highly-specialised vertical applications for mass billing, finances and accounting, monitoring of accounts receivable, material and warehouse management, business analytics, and prediction of future behaviours of consumers are suitable for the most challenging environments. In 2019, they had issued more than 5 million of bills to Slovenian households for their energy consumption, water supply, and several utility costs.
- Brief description of the project and its goals
    - The main objective of the project is to test how the 3TAV's existing code for energy consumption prediction scales form their local server to a small or medium size supercomputer.
    - They have developed Python and R scripts to retrieve data, store it to MongoDB and load it back when needed. Additionally, based on the historical data they have developed scripts that build prediction models for each consumer and make prediction for the 24 hour consumption (on the 15 minute granularity) for day d+1 based on his history up to day d-1.
    - Using deep neural networks, building each data model takes approx. 2 minutes and approx. 8 MB of memory. They want to have a solution that is capable to build approx. 100,000 models in few hours and store them to HDD memory.
    - Therefore, the main goal of the project is to test how they can adapt the existing scripts such that they can build 100,000 models and predictions within time limit of approx. 20 hours using a medium size supercomputer with state-of-the-art computing nodes and storage.

Activity performed

- Created container with MongoDB database and tested and deployed it on HPC at UL FME
- Moved the mongo DB to HPC at UL
- Created tasks that enable shared memory paralelisation using the library foParallel, parallel and foreach;
- Performed comprehensive testing on one compute node that show libraries with scripts parSapply and parLapply and foreach - dopar work very well, scaling is close to linear, if computing models (time critical operations) is done in parallel;
- MPI parallelisation is under study.

PRACE cooperation

SME attended PRACE 2 training events organised by Slovenia's PTC: training about big data technologies

Benefits for SME

- Knowledge about new technologies;
- Possible solution how to overcome difficulties related to too long computational times of their software
- Proof of concept how to continue development of their solution;
- New ties with HPC centres and academia for future development of HPC and Big Data technologies;

Lessons learned
The project is in progress

3.5.8 Voxo (Sweden)

Project Partners
SME contact: Johan Wadenholt
PRACE contact: Mark Abraham (ENCCS)

Overview
This project aims to develop a Swedish-language speech-synthesis model. Its partners are the Swedish SME Voxo AB, and the EuroCC National Competence Center of Sweden (ENCCS) acting in the role of a PRACE advisor.

Voxo specialises in extracting, analysing, and visualising voice data. Their services are used in multiple industries where they provide insights and enable data-driven business development. Voxo is keen to build on existing voice expertise to enter market sectors that need the capability to synthesise Swedish voices. The generation must not compromise the integrity of the data, which might be personal to a user. Thus, existing programmatic APIs are unsuitable and they need to build our own solution using HPC.

They intend to use machine learning to develop a proprietary Swedish-language speech-synthesis model. It will be a key component of a conversational assistant capable of providing information in real time in response to spoken natural-language questions. It needs to be capable of learning to pronounce jargon relevant to particular domains, such as banking. Ideally, a model could learn to copy a user's pronunciation of their name. It will generate audio streams quickly. Users will be much more comfortable if the conversation flows naturally, without pauses for generating long replies. It will be implemented using existing Tacotron and WaveGlow technology, such as described in this blog post [27]. Voxo has already prototyped such a model, but needs to refine the process for Swedish and generalise to other Nordic-region languages.

Activity performed
We first applied for computer time on JUWELS Booster and were successful there.

In the application we referred to the GitHub repository maintained by Nvidia containing the Python project that they used for training Tacotron and WaveGlow models. Between planning the project and starting the work, the support activity there waned. We attempted to get that framework running on JUWELS Booster. Only one version of CUDA was available, which did not suit the versions of some of the older Python packages required. Once we had solved that, it was hard to get the Python packages to find binaries for e.g. audio-file conversions from the EasyBuild stack installed on JUWELS. We also attempted to use the JUWELS container-building infrastructure, but it had a bug (which was reported to JUWELS). Instead, we attempted to build our own Docker/Singularity containers. Those worked locally but not on JUWELS Booster. Over time, about two person-weeks of effort was spent by ENCCS in those attempts.

Speech synthesis is an extremely fast-moving space. New research and ML frameworks emerge all the time. We decided to give up on that older repository and seek a new way to train such models. We identified a GitHub project called TensorFlowTTS that would train the kinds of models that we targeted, was being actively supported, and claimed multi-GPU training

support. This package, too, had problems to work around in order to run in an HPC space. For example, one cannot download content from a process running on a JUWELS login node, but the package has a large range of functionality that is not needed for Voxo's SHAPE project whose initialisation routines require such downloads. Those downloads had to be identified and either satisfied with a local file, or disabled in the Python source code. A suitably crippled version was run on JUWELS Booster, but the performance on multiple GPUs was the same as on a single GPU. As the queue system on JUWELS Booster only allocates whole nodes, we reached the point where we would either have to be very inefficient with our allocation or design multiple training runs to run concurrently on the separate GPUs. Alternatively, we could attempt to debug why the training process did not scale (e.g. bogged in CPU-side pre-processing of many small audio files).

Instead, we identified that Nvidia's effort had clearly pivoted to the NeMO framework [28] which looks like an excellent implementation suitable for this project running on JUWELS Booster. We are still in the early stages of deploying this framework to train with Voxo's data.

PRACE cooperation

Naturally JUELICH staff were involved in support requests, for which ENCCS and Voxo are thankful – these were excellent and timely responses.

ENCCS expended essentially all the effort in the project so far, because the agreed division of effort would be for ENCCS to teach Voxo how to use the infrastructure once it was working. ENCCS applied for the computer time on JUWELS Booster. ENCCS expended all the effort on attempting to get ML frameworks working successfully at scale. As that has not yet worked in a useful way, Voxo is still waiting to apply its expertise to training the models.

Benefits for SME

Voxo has been developing a conversational agent for personal banking in the Swedish market. This will make personal banking much more accessible, by removing requirements like visiting a branch, or having a mobile computing device.

Users will ask spoken-language questions like "How much money do I have in my accounts?" Once the agent has understood a user's intent, it may need to consult data sources and synthesise a reply in real time. There are existing API-based products for multi-lingual speech synthesis, such as Amazon's Polly. However, the personal data of European users must be handled in accordance with the GDPR, e.g. not leaving Europe. Having a text-to-speech solution in-house at Voxo ensures such compliance.

Success with this project makes it possible to bring such a product to the market. Advantages that come from the proposed approach are

- understanding and control of the model-development process de-risks products built with it, e.g. because it can train on correct pronunciation of jargon
- understanding the requirements and limitations of the model-development process makes it easier to plan to build models in new languages
- the input of PRACE HPC expertise will prepare Voxo to use HPC efficiently now and independently in future
- more languages scale the technology to more markets for stronger business impact to Voxo

As this project is still ongoing, no benefit has yet accrued.

Lessons learned
In applying it was difficult to understand whether a successful SHAPE application automatically obtains machine access. This was a particular problem because the PRACE centre supporting the project (i.e. ENCCS) did not have suitable local hardware to use. However, it was not a problem in practice, once JUELICH agreed to provide computer time to the project.

Otherwise, considerable time needs to be allocated for getting non-HPC frameworks running on HPC resources, and future projects should have that advice available to them.

3.5.9 MultiplexDX, s.r.o. (Slovakia)

Project Partners
SME contact: Peter Kilián
PRACE contact: Michal Pitonak, Lukas Demovic (CC SAS)

Overview
The SME is a biotech company working in cancer diagnosis and wishes to utilise multicore systems to deal with the large amount of RNA data.

Activity performed
1. The SME in cooperation with PRACE managed to get the whole software stack installed on CCSAS's HPC cluster.
2. Initial testing and validation runs on synthetic data were performed
3. Several auxiliary scripts to enable a data pipeline between various (mostly R) programs were coded by the SME and optimised by PRACE.
4. Initial set of testing data (~300 GB) from publicly available repositories (using NCBI SRA - Sequence Read Archive - tools) were downloaded using custom scripts.

PRACE cooperation
PRACE was involved in each stage of the project that was carried out so far. At the initial stage the PRACE representatives described their local HPC environment and proposed a strategy for scheduler integration and orchestration of the rather complicated set of software tools required by the SME. In the next stage, PRACE assisted in installation of the software stack that was quite diverse (mostly Python and R programs / libraries), as well as in writing and optimising several auxiliary orchestration scripts. In the last stage we recommended an optimal strategy for transferring of large amounts of data to our HPC site and assisted in its implementation.

Benefits for SME
The SME obtained qualified support in migrating their complicated workflow to the HPC environment thus enabling access to unprecedented computing power. Code optimisation hints, already at this "pre-production" stage, were valuable and increase throughput of data processing notably.

Lessons learned
The project is in progress though may not be completed due to change of staffing within the SME.

3.5.10 BuildWind SPRL (Belgium)

Project Partners
SME contact: Alessandro Gambale
PRACE contact: Bertrand Cirou (CINES)

Overview
BuildWind uses advanced, three-dimensional numerical simulation to predict airflow, heat transfer, and contaminant transport inside and around buildings. We assist city planners and building designers in creating healthier, safer, and more sustainable urban and industrial environments

Some preparatory work will be necessary in the beginning, in order to create geometry and mesh, create OpenFOAM (CFD software) script and define simulation parameters. Then the software will be handed to the PRACE partner for installation and possible code optimisation will be proposed. To test the framework, about 500 simulations will be run first and examined, before proceeding with the remaining cases, up to a total of about 3500-5000 simulations. Then, after processing the data obtained from the simulations, a machine learning algorithm will be used to generate a predictive model for urban environmental assessment of the simulated site.

Activity performed
- Search for an appropriate supercomputing resource among PRACE partners.
- Account creations for the team on ARCHER2.
- Mesh and OpenFOAM code modifications done.

PRACE cooperation
Chris Johnson managed to get CPU hours on ARCHER2

Benefits for SME
- The UrbEM project would allow us to make short simulation time, so that we could fully validate and demonstrate our technology during the CELTIC-NEXT SPICECO project we are participating and promoting it together with other services for smart-cities.
- If the validation process of UrbEM is successful, we expect to hire an additional 2 members to work full time on the technical and business development of combined AI and CFD applications for smart-cities and to generate at least 50 -100 k€ revenues already in 2022, which corresponds to a 15-30% increase in our expected turnover.

Lessons learned
Project is just starting

3.5.11 Reintrieb GmbH (Austria)

Project Partners
SME contact: Robert Kastl
PRACE contact: Bernhard Semlitsch, Claudia Blaas-Schenner, Harald Grill (TU Wien)

Overview
Reintrieb GmbH is a clean-tech R&D company, specialising in mechanical engineering of clean gear and propulsion systems in the marine and inland water ways sector. Reintrieb wants to bring two patented products on the market. An oil-free water-lubricated high-performance gearbox and the side-by-side (SBS [EP 2613999]) drive relevant for this project.

The SBS drive consists of two counter-rotating propellers that are arranged next to each other in a nozzle. The aim of the SBS propulsion system is to reduce the draught in inland navigation for the inland shipping industry that is severely hit by climate change. With the SBS drive vessels can be operated at lower water levels. The feedback from the market / potential

customers was that the SBS drive must be at least as efficient as conventional large propeller drives.

By using two small propellers, the same thrust can hopefully be realised as with one large propeller with a flatter drive arrangement. An increase in thrust (and thus efficiency) is to be achieved by using counter-rotating propellers so that the counter-rotating swirl components can balance each other out. In the previous development steps empirical data has already been collected with air and open-water tests and the underlying theories for draught reduction and drive efficiency have been confirmed.

The aim of the project is to fundamentally model the effect of the various influencing variables on the efficiency of the side-by-side drive by means of CFD calculations and, consequently, to calculate an optimal propeller arrangement and nozzle shape. These influencing variables are: The distance and angle of the propellers to each other, the nozzle closure around the propellers, the nozzle shape, the nozzle resistance at speed, the suction effect of the propulsion system and the change in inflow due to the hull shape of the ship and the clearance under keel.

Activity performed

The key challenge of simulating closely arranged propellers is handling the rotation with respect to each other. The state-of-the-art approach is to utilise separate volumes for each rotating and non-rotating domain. Therefore, the rotating components of the geometry are discretised in cylindrical volumes, which are embedded in the non-rotating domain. The cylindrical volumes, representing the discretisation fixed around the propeller, are then rotated at each time step. The flow information must be transferred at the bounding faces of these cylindrical volumes, which are two-dimensional interfaces.

The close arrangement of the two propellers does not allow this approach because the two cylindrical volumes (one for each propeller) would intersect with each other. Thus, the overset approach must be employed. A body-fixed discretisation is used for each propeller, which overlaps with a ‘background’ discretisation. Only the near-surface effects are modelled on the body-fixed discretisation, while the large-scale flow phenomena are resolved on the background discretisation. The flow information is passed via a volumetric interpolation at each time step. Thus, the computational requirements are much higher, but even discretisation intersections can be simulated.

Initial propeller simulations have been performed. To enable the verification of the overset approach, the single propeller is considered with both approaches, i.e. the state-of-the-art approach and the overset approach. The discretisation of the propeller has been done using SnappyHexMesh of OpenFOAM. The first simulations show the general simulation procedure is working, but quality improvements of the discretisation are required to deliver better comparison with experimental data.

PRACE cooperation

EuroCC Austria approached Reintrieb with the aim to get a practical industry example for a complex CFD, but it turned out that Reintrieb had problems with its CFD research partner and therefore the CFD had not been done yet. Therefore, EuroCC Austria got to know the problem of Reintrieb and approached the Austrian PRACE-6IP partners. They teamed up and together had several meetings with Reintrieb to clarify all the framework conditions and details with Reintrieb. Reintrieb got the full support from information about possibilities for writing a PRACE SHAPE proposal from the Austrian EuroCC and PRACE partners.

PRACE provides the necessary person months for the CFD/HPC expert Bernhard Semlitsch of TU Wien, while the HPC resources of VSC-4 are provided directly by the VSC Research Center of TU Wien.

Benefits for SME
The expectation is that HPC enabling resulting from this project will significantly reduce time-to-market and substantially reduce costs for the otherwise customary test loops and iterations of drive models in open water basins tests. The results of the CFD calculations will be checked in model tests and can then be verified with a full-size prototype in real-life operation. Reintrieb GmbH has already found a propulsion drive manufacturer and a ship operator to build and test the prototype.

The prototype (side-by-side drive) is to be completed before the end of 2021 and tested by Q2 2022. Thereafter, series production and sales of 400 SBS drives are planned until 2025. In addition to the SBS drive, a simple side-by-side propeller unit for retrofitting conventional shaft drives will be developed.

Lessons learned
The project is not (and was not planned to be) that far evolved in the middle of October 2021. For the time being there are no lessons learnt that can be shared with the public.

3.5.12 SIRIS Academic S.L. (Spain)

Project Partners
SME contact: Francesco Alessandro Massucci
PRACE contact: Marta Villegas, Aitor Gonzalez-Agirre (BSC)

Overview
SIRIS Academic S.L. is a research-intensive consulting company, based in Barcelona, focused on policy making in the science, technology & innovation (STI) sector. SIRIS has worked with more than 100 institutions (universities, research institutes, local governments, foundations) in more than 25 different countries.

The objective of the project is to train and apply a series of textual classifiers on a large number (tens to hundreds of millions) of STI documents (scientific publications, research projects, patents), by leveraging the latest advances in natural language processing (NLP) methods. The project aims to harness High-Performance Computing platforms on large scholarly textual corpora in order to scale up the analysis capability and bring it to an entirely new level.

Activity performed
The main idea is to present different avenues of possible technical solutions, and then choose one of them and explore it in detail, pushing it to production level.

The project is divided in three phases:

1) Pilot: SIRIS Academic will provide a working example of a dataset that has already been studied and curated to some degree of accuracy and detail. With a carefully controlled system, we can proceed with several important preliminary tasks:
    - Set up the infrastructure. Create users in the HPC cluster; deliver analysis code; install necessary libraries for performing analyses.

- Small tests. Run existing codes within the HPC system; benchmark different kinds of architectures (single-node CPU, vs multiple-node CPU, vs single-node GPU, vs multiple-node GPU) for the different computational tasks.
- Model assessment. With the chosen model (see below) perform full model prediction runs on the testing dataset, and evaluate its performance.

2) Production: Once the results of the pilot phase are in, we will then proceed to using more computational power to analyse larger data sets. In this phase we intend to carry out the following tasks:
   - Preparation of datasets. In this phase the large-scale datasets will be downloaded and deployed on the file systems of the BSC.
   - Initial testing phase. Here we will fine tune the parameters of the runs, benchmarking the requirements in terms of time and memory of the codes to run.
   - Test runs. In this phase we will launch the test production runs, where data will be generated to be analysed.
   - Data analysis. Here, we will look at the results of the runs and assess their performance.
3) Reporting: At the end of the production phase we will write a synthetic report with the details of the results that we obtained, and the main conclusions in terms of model performance and optimal setup of the HPC system.

PRACE cooperation
The cooperation includes coaching and guidance by the BSC team, but not access to machines. Nevertheless, the BSC team is open to run some of the preliminary experiments that will allow to choose the best approach for the project.

Benefits for SME
The impact of the project could be significant for SIRIS Academic Business. To construct analytical reports for their clients, they invest large amounts of time to build accurate representations of thematic research areas, in the form of controlled vocabularies that allow them to filter documents by detecting specific keywords (e.g. in the domain of “Climate change”, the keyword “Ocean acidification” would allow to detect relevant documents).

However, this approach is very time-consuming, whereas new methods based on textual classifiers leveraging deep learning would allow for a far more scalable detection of relevant documents. Running deep learning methods (pre-trained models as well as training our own) on a large corpus of documents would help the SME to save the time dedicated to the creation of vocabularies and to obtain richer results, making them more competitive.

Lessons learned
The project is still at a very early stage.

3.5.13 Ingénierie et Systèmes Avancés (France)

Project Partners
SME contact: Philippe Reygnier
PRACE contact: Isabelle Dupays and Laurent Leger (IDRIS)

Overview
Ingénierie et Systèmes Avancés (ISA) has a core capability in computational fluid dynamics, this for a large range of flow regimes. They can be simulated using 2D/3D codes based on the finite volume approach and structured or unstructured meshes.

The use of the HPC infrastructure will allow the prediction of aerodynamic and aerothermodynamic databases in a shorter time when comparing to the use of the in-house capabilities. The second benefit will be the external support from experts in order to increase the parallel efficiency of the tools. This particularly applies to the set-up of an ad-hoc script in order to perform the computations in an automatic way. For these reasons the support of SHAPE will be an important asset for the company in order to increase the computational efficiency. The project will benefit from this support, but this will also enlarge the capabilities of ISA to description of the project and its goals.

Activity Report
The SU2 code uses by the PI has been installed on the Jean Zay machine with MPI, Intel compilers and MKL library. SU2 is an open-source collection of software tools written in C++ and Python for the analysis of partial differential equations (PDEs) and PDE-constrained optimisation problems on unstructured meshes with state-of-the-art numerical methods. The software is available on a git repository. The PI has provided us some simple tests cases to validate the installation of SU2. The next phase will be to test the scaling of the code on a realistic case and produce a profiling to see the bottlenecks.

PRACE cooperation
Experts who helps the PI, access to IDRIS machine Jean Zay, HPE SGI 8600 supercomputer, with a budget of 50,000 CPU cores hours.

Benefits for SME
Business test case, start to work on HPC and supercomputer.

Lessons Learned
The project is just starting.

3.6 SHAPE 14 and future calls

The SHAPE 14 call opened on 1st October 2021 and closed in 15th Nov 2021 receiving a record 19 proposals. The panel will decide by the end of 2021 on which proposals to recommend for award with successful projects from this call running during the extension period (Jan – Jun 2022). SHAPE 14 was the final call under PRACE-6IP with future SME interaction likely to happen under EuroHPC. During the PRACE-6IP extension on-going projects will continue to run and SHAPE 14 projects will start. During this time discussions will take place with the EuroHPC project on the best way to continue SHAPE-like activities into the EuroHPC era.

4 T7.3 DECI Management and Applications Porting

This section describes the current status of DECI with an overview given in Section 4.1 and Section 4.2 describing the most recent calls. The DECI-17 call was the final call under PRACE-6IP and in Section 4.3 we discuss DECI future as we move into the EuroHPC era.

4.1 Overview of DECI

The PRACE Distributed European Computing Initiative (DECI) provides access to Tier-1 level resources. DECI began under Distributed European Infrastructure for Supercomputing Applications (DEISA) in 2005 with 17 calls having been launched awarding over 1.6 billion core hours. Since then it has run within a PRACE Implementation Phase (PRACE IP) project, followed by a spell as a PRACE Optional Programme and more recently it has again run within a the WP7 work package of the PRACE 6-IP project. To date, DECI has received 1223 proposals resulting in 729 projects and the programme remains popular with the most recent three calls receiving 73, 68, and 81 proposals respectively from 28 different countries (including 41 proposals from 8 of the EU13 countries). The processes for DECI have been described in [4] and won't be repeated here.

4.2 DECI Programme Status and Project Enabling

4.2.1 DECI-15

Projects from this call ran between June 2018 and June 2019 and the 40 awarded projects have now finished, apart from one project given a long extension locally at PSNC.

4.2.2 DECI-16

The DECI-16 call was opened on 16th December 2019 and closed on 31st January 2020 for projects running from June 2020 to June 2021. The call received 68 proposals and awarded ~560 M DECI standard hours (~160 M machine core hours) of resources overall across 48 projects. Of these, 12 projects (~122 M DECI standard hours (~35 M machine core hours)) were from countries not contributing resources: Belgium (2), Bulgaria (1), Switzerland (1), Cyprus (2), Denmark (1), France (1), Latvia (2), Serbia (2). The full list of projects awarded from the call were listed in [4]. Most projects have now finished apart from a few given extensions.

4.2.3 DECI-17

The DECI-17 call opened on 16th December 2020 and closed on 31st January 2021 for projects running from June 2021 to June 2022. The call received 81 proposals and awarded ~886 M DECI standard hours (~250 M machine core hours) of resources overall across 55 projects. Of these, 17 projects (~27%, or ~235 M DECI standard hours (~67 M machine core hours)) were from countries not contributing resources: Switzerland (1), Germany (2), Denmark (1), Spain (4), Italy (6), Slovakia (1), Turkey (2). The full list of projects can be seen in Table 11 and the list of systems the projects were awarded on is given in Table 12.

DECI-17 Project Acronym	Country of PI	DECI Home Site	Subject Area	DECI standard hours awarded
FRECUENCIA	Czech Republic	VSB-TUO	Materials Science	24,325,000
LipidDyn	Denmark	CSC	Bio Sciences	5,940,000
parGASCIP	Germany	EPCC	Materials Science	8,000,000
SCAPHOL	Germany	EPCC	Materials Science	23,077,600
CFDROAD	Greece	GRNET	Engineering	7,611,940
AccurateQChem	Hungary	KIFU	Bio Sciences	10,920,000
HeatS	Ireland	ICHEC	Materials Science	11,700,000
NcpT2D	Ireland	ICHEC	Bio Sciences	10,399,999
APPSEI	Italy	CINECA	Materials Science	13,464,000
BLAVO	Italy	CINECA	Engineering	9,800,000
SOLID	Italy	CINECA	Engineering	22,464,000
SSMBHB	Italy	CINECA	Astro Sciences	11,232,000
StarCluBin	Italy	CINECA	Astro Sciences	7,520,000
TuSSQbPro	Italy	CINECA	Materials Science	6,959,160
ACCRETING	Netherlands	SURF	Astro Sciences	10,560,000
ANEMONE	Netherlands	SURF	Engineering	10,436,000
BHSLSE	Netherlands	SURF	Astro Sciences	18,400,000
DRAG	Netherlands	SURF	Engineering	18,800,640
MechICat	Netherlands	SURF	Materials Science	19,998,400
TangoSIDM	Netherlands	SURF	Astro Sciences	20,000,000
VisIM	Netherlands	SURF	Applied Mathematics	20,736,000
BHAO4	Norway	Sigma2	Earth Sciences	8,712,000
Coolcarbon	Poland	TASK	Plasma & Particle Physics	6,300,000
DYNNETOPT	Poland	WCSS	Informatics	8,750,000
FPDYNAMICS	Poland	WCSS	Materials Science	33,950,000
MicroGravityPHP	Poland	WCSS	Engineering	33,000,000
MOLMABIM	Poland	PSNC	Materials Science	20,000,000
NERVOMOLSIM	Poland	PSNC	Bio Sciences	57,600,000
PostTransRNAMod	Poland	PSNC	Bio Sciences	9,600,000
AbSpin	Portugal	UC-LCA	Materials Science	7,514,430
BHBF	Portugal	UC-LCA	Astro Sciences	7,000,000
COIMBRALATT7	Portugal	UC-LCA	Plasma & Particle Physics	28,000,000
NANOREAL	Slovakia	CCSAS	Materials Science	20,800,000
AntiViRNA	Spain	CESGA	Bio Sciences	1,400,000
GRSimulations	Spain	BSC	Astro Sciences	19,008,000
MECHANOMACB	Spain	BSC	Bio Sciences	12,860,000
MEGAWAVES	Spain	CESGA	Plasma & Particle Physics	12,792,000
AMCVD2	Sweden	PDC	Materials Science	14,399,200
CHAIN	Sweden	PDC	Engineering	15,998,400
EnginZyme	Sweden	PDC	Bio Sciences	4,818,487
H2CFD	Sweden	PDC	Engineering	10,000,000
LSIF	Sweden	PDC	Engineering	6,800,062
MANOSODE	Sweden	PDC	Materials Science	20,001,600
NANOCAT	Sweden	PDC	Materials Science	13,992,000
P2021CEO2SB	Sweden	PDC	Materials Science	22,007,162
Q2Dtopomat	Sweden	PDC	Materials Science	24,003,000
DEFDOW	Switzerland	UHeM	Materials Science	13,412,648
EnDy	Turkey	UHeM	Bio Sciences	20,946,534
EPICENTROMERE	Turkey	UHeM	Bio Sciences	24,960,000
ASCOTEPST	UK	EPCC	Plasma & Particle Physics	4,252,560
ESTONT	UK	EPCC	Engineering	14,137,344
GRAPHSAC	UK	EPCC	Materials Science	18,676,736
LongtermCFI	UK	EPCC	Plasma & Particle Physics	13,062,400
radfeedback	UK	EPCC	Astro Sciences	53,040,000
RatCat	UK	EPCC	Materials Science	12,250,000
			Total	**886,389,302**

Table 11: The list of DECI-17 projects awarded

System	Architecture	Site, Country
Mahti	Atos AMD Rome EPYC 7H12 @ 2.6 GHz	CSC, Finland
Prometheus	Intel Xeon E5-2680v3 @ 2.5 GHz	CYFRONET, Poland
ARCHER2	Cray dual AMD 64-core @ 2.25 GHz	EPCC, UK
ARIS	IBM NextScale Ivy Bridge @ 2.8 GHz	GRNET, Greece
ARIS	DELL PowerEdge dual NVIDIA K40 @2.8 GHz	GRNET, Greece
ARIS	DELL PowerEdge dual INTEL Xeon Phi 7120p @ 2.8 GHz	GRNET, Greece
Kay-thin	Intel Xeon Gold 6148, Skylake 2×20-cores @ 2.4 GHz	ICHEC, Ireland
Kay-Fat	Intel Xeon Gold 6148 Skylake, 2×20 @ 2.4 GHz	ICHEC, Ireland
Kay-GPU	NVIDIA Tesla V100 16GB PCIe, Volta architecture	ICHEC, Ireland
Kay-Phi	Intel Xeon Phi Processor 7210, Knights Landing @ 1.3 GHz	ICHEC, Ireland
Leo	Ivy-Bridge Intel Xeon E5-2650v2-core @ 2.6 GHz + Nvidia K20x, K40x	KIFÜ, Hungary
DARDEL	AMD dual EPYC 2.25 GHz 64 core	KTH-PDC, SNIC, Sweden
EAGLE	Intel 8268 @ 2.9 GHz,	PSNC, Poland
EAGLE	GPU Nvidia V100	PSNC, Poland
Saga	Dual Intel Xeon Gold 6138 Skylake @ 2.0 GHz	Sigma2, Norway
Betzy	Dual AMD Epyc 7742 @ 2.25 GHz	Sigma2, Norway
Snellius	AMD Rome, EPYC 7H12, 2x64core @ 2.6 GHz	SURF, The Netherlands
Navigator	Intel Xeon E5-2697 v2 @ 2.76 GHz	UC-LCA, Portugal
Barbora	ATOS 2x Intel Cascade Lake 6240, 18-core @ 2.6 GHz	VSB-TUO, IT4Innovations, Czech Republic
Bem	Intel Xeon E5-2670v3 @ 2.3 GHz (Haswell)	WCSS, Poland

Table 12: The list of systems DECI-17 projects were awarded hours on

4.2.4 DECI Statistics

In Figure 45 we show the range of countries applying to, and receiving projects from, DECI. This is shown for the last three calls so includes statistics from 222 proposals and 143 projects. As can be seen the range of countries receiving DECI awards is large and diverse.

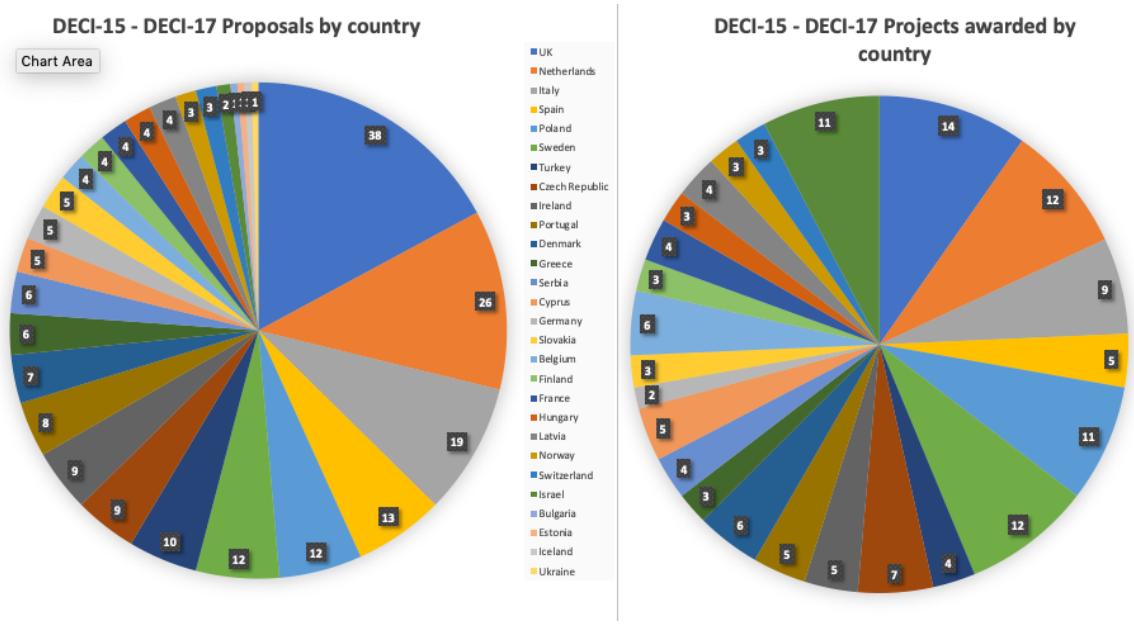


Figure 45: DECI proposals received and projects awarded by country cumulatively for DECI-15 - DECI-17

Figure 46 then shows the subject areas for the same group of proposals and projects, showing that as expected this is dominated by the physical sciences, followed by engineering.

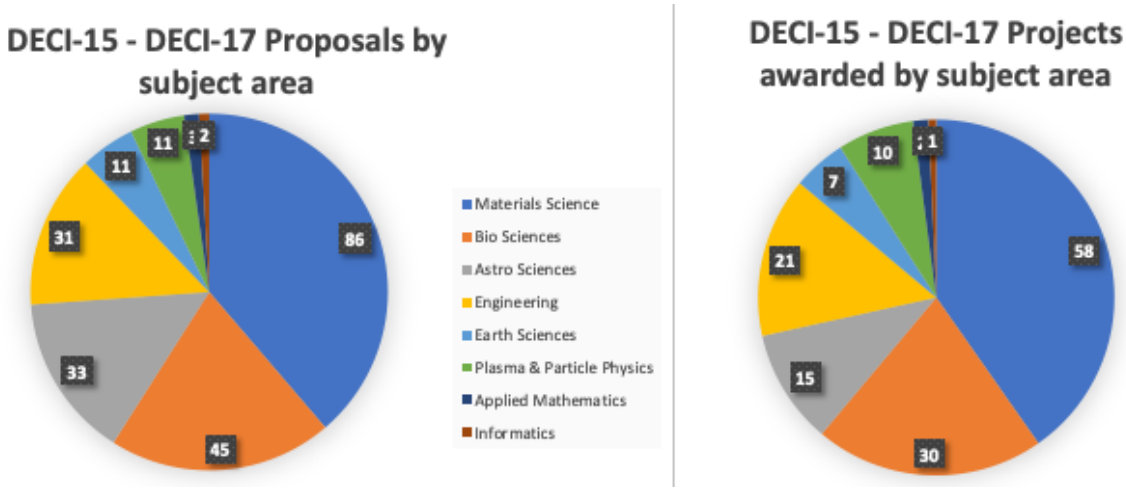


Figure 46: DECI proposals received and projects awarded by subject area cumulatively for DECI-15 - DECI-17

4.2.5 Enabling work

For the most recent calls (DECI-16 and DECI-17) a small amount of effort was available to carry out enabling work. Most of the effort is used to provide fairly routine help, such as giving advice on use of batch queuing system, compiler usage, etc. and setting up accounts for users in all of the projects which each typically have 1-4 users. Sometimes more assistance is required in installing specific versions of software. For example, Nek5000 could not be compiled using newer versions of the GCC compiler (v9.x or later) with default flags. The team at SNIC, PDC-KTH helped the user of the CHAIN project to fix the issue by adding the relevant flag as well as fixing other small problems using the Cray CCE compiler. This enabled the users of this project to run Nek5000 built with both the GCC and Cray CCE compilers on ARCHER2.

4.3 DECI Future

The DECI-17 call was the last call to be issued under PRACE-6IP. Projects for DECI-17 were given from June 2021 to June 2022 to carry out computations. As the PRACE-6IP project finishes in June 2022 any projects given extensions will have to be negotiated on a case-by-case basis with the participating sites. Any final reports can still be collected as the effort required to process these is minimal.

As can be seen the calls remain popular with researchers across Europe. As DECI provides a unique way for resources to be exchanged across Europe it is hoped that DECI can be taken forward into the EuroHPC era in some form. A paper is in preparation which explains the benefits of DECI to European researchers. This paper will be made available with the current set of deliverables.

5 T7.5 Enhancing the High-Level Support Teams

T7.5 is a new task implemented in PRACE-6IP. It will provide additional effort to enhance the work of the PRACE High-Level Support Teams (HLSTs). The HLSTs will enhance the scientific output of the Tier-0 systems through the provision of level 3 (midterm activities) support activities. Task 7.5 supplements those activities and extends this work with specific expertise from the other PRACE centres. This ensures sharing of expertise across PRACE to maximise the benefits to users of the PRACE systems. Task 7.5 will work with the HLSTs to extend and enhance their activities supporting Tier-0 users, and also Tier-1 intensive users targeting Tier-0, in order to maximise the scientific output of the Tier-0 systems.

As a new activity within this work package, the first steps were to define and clarify the scope of this task. As the task definition was to offer level 3 support to "large" users of the PRACE platforms, we agreed on the fact that specific acceptance criteria, as well as a very well defined project process are the key to enhance the global throughput of the systems.

Once the basis was set, the first projects were able to start.

The main concern for this activity, and to make sure that there is no overlap with other activities (such as Preparatory Accesses), was to make sure that we provide a level 3 support such as:

1) HPC Co-development (specific libraries, new architectures, …)
2) Enabling workflows (HTC requirements, dataflow management, …)

All the projects were performed with the objective to optimise the global throughput of the applications and hence maximising an efficient usage of the HPC resources in Europe.

5.1 Project acceptance criteria

The criteria for project acceptance are the following:

1) Scalability: targeting a scalability that only a Tier-0 platform can provide
2) Architecture: HPC hardware targeted not accessible on Tier-1 platforms
3) CPU hours volume: exceeding what could reasonably be provided on a Tier-1 platform
4) Memory/storage requirements: exceeding what could reasonably be provided on a Tier-1 platform

On top of that, a maximum of 2 projects per partner will be accepted (as only 10 PMs are available per partner).

5.2 Project process

A clear process was agreed with all partners within this task during the Face-to-Face meeting in Antwerp (October 2019).

This process includes the following points:

1) Partners contact users that are using a huge fraction of their available system to offer them level 3 support
2) Users answer the partners with specific project and technical issues to be addressed
3) A monthly teleconference
4) Project selection during the teleconference based on the criteria
5) If the project is accepted, then:

   a. The project starts on local Tier-1 system
   b. Apply for a Preparatory Access on targeted Tier-0 system
   c. Move to Tier-0 and validate the developments
   d. Collaborate with local HLST to validate the developments
6) Activity reporting

5.3 Activity report for each partner

5.3.1 EPCC: Optimisation and tuning of NAMD and NEMO to prepare users for Tier-0 systems

Overview of the project:
This project investigates the optimisation and tuning of NAMD and NEMO to advise users on maximising the scientific output of resources when moving from Tier-1 to Tier-0 systems [29][30]. We provide advice based on process placement and runtime configurations to optimise for different architectures. Investigations were conducted on the UK national Tier-1 system ARCHER2, an HPE Cray EX Supercomputer operated by EPCC which will contain 5,860 compute nodes, each with two 64-core AMD EPYC 7742 processors and 8 NUMA regions per node [31]. The current ARCHER2 service is provided by a 4-cabinet preview system with 1,024 compute nodes. Optimisations based on ARCHER2 are suitable for aiding users moving from Tier-1 to Tier-0 as it is an example of modern trends in supercomputing and representative of current and future Tier-0 with a many cores per node architecture. PRACE Tier-0 Joliot-Curie is selected for further testing because the processor architecture is very similar to ARCHER2: Joliot-Curie has a BULL Sequana XH2000 compute partition with 2,292 compute nodes consisting of two 64-core AMD EPYC processors each [32]. Further performance tests based on Joliot-Curie are ongoing.

Test cases from the PRACE UEABS repository were used to investigate the performance of NAMD and NEMO [33]. Both are standard benchmarks suitable for optimising computational performance on Tier-0 systems but have limited application: typical production runs for NAMD have high memory requirements and NEMO performs significant I/O management [34][35]. Despite this, our investigations successfully find that NAMD users should match the architecture of NUMA regions to find the optimal balance of thread and process parallelism. Reserving a thread per process improves the scalability on ARCHER2 by 12.5%. NEMO users can increase the efficiency of jobs by confining XIOS IO servers to NUMA regions and run larger simulations by reserving 75% of an ARCHER2 node without a performance detriment.

Presentation of the work performed during the project:
Figure 48 shows the strong scaling of NAMD to demonstrate the linear speed-up of large test cases up to 16,384 cores. Figure 49 investigates the balance between thread and process parallelism on 4,096 cores; the number of threads per process is chosen such that nodes are always fully populated with either threads or processes. Figure 49 shows the best performance is found without threaded parallelism with 128 processes per node and 1 thread per process; this is expected for our simplistic test case due to the additional communication overheads with threads. However, most users will require threaded parallelism to reduce high memory requirements by introducing multiple threads per process which share the data structure stored by the parent process [34]. The optimal balance of threads and processes is found with 16 processes per node and 8 threads per process. This matches the ARCHER2 NUMA structure, with 8 NUMA regions per node and 16 cores per region, mapping 2 processes per NUMA

region to benefit from the memory hierarchy [31]. This is observed for 8 and 28 million atoms whereas 210 million atoms required less processes to further reduce memory requirements and still map 1 process per NUMA region. Hence, users should match the architecture of NUMA regions to find the optimal balance of thread and process parallelism.

The NAMD developers recommend reserving a thread per process for communication on systems with a many cores per node architecture to reduce communication between threads [34]. Figure 47 shows that reserving a core improves scalability by 12.5% as the turning point is delayed by 8,192 cores. At lower core counts the performance benefits from the additional computation threads, as seen in Figure 47. NAMD users should reserve a thread per process for communication to improve the scalability.

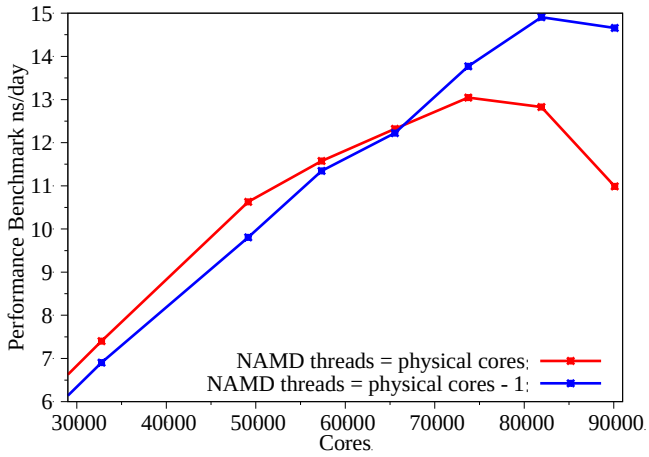


Figure 47: Strong scaling of NAMD simulating 8 million atoms with 32 processes per node and 4 threads per process scaling up to 90,112 cores.

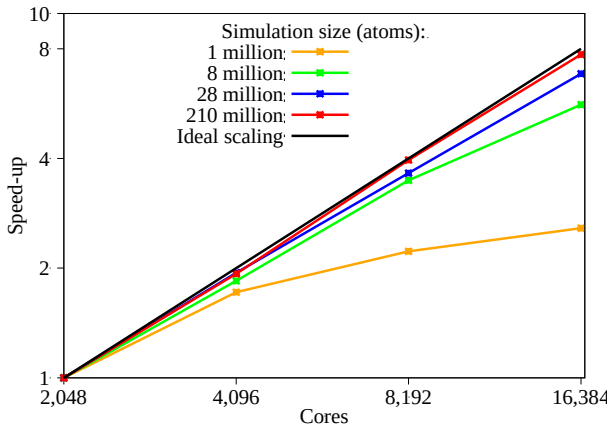


Figure 48: Strong scaling of NAMD simulating 1, 8, 28 and 210 million atoms. Nodes fully populated with 128 processes and 1 thread per process.

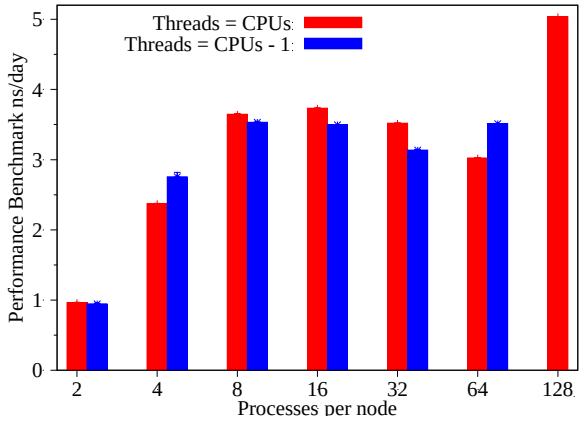


Figure 49: Investigating NAMD by varying the balance between thread and process parallelism with 8 million atoms on 4,096 cores.

The NEMO developers recommend the use of detached mode, where each process runs as an XIOS client with external XIOS I/O-server processors, to efficiently manage I/O when performance is limited by large volumes of data [35].

Figure 50 investigates the regular packing of ocean clients and 0, 1 or 3 idle cores between each client to observe performance when under populating ARCHER2 nodes. Due to the limitations of the chosen test case, we can run with minimal XIOS I/O server cores and expose the effects of under population. Figure 50 shows that NEMO's computation is unaffected despite only populating 25% of the node, allowing users to reserve large portions of ARCHER2 nodes for I/O management without a performance detriment. Figure 51 extends this investigation by increasing the number of XIOS I/O server cores per node whilst fully populating the remainder of the node with clients. We observe limited variation in performance for 4, 8, and 16 cores per XIOS I/O server, as expected due to the limitations of this test case and the results in Figure 51. However, when allocating cores that are outside of a single 16-core NUMA region, we begin to see a detrimental impact on performance. Users should confine XIOS I/O servers to individual NUMA regions to benefit from the memory hierarchy.

Outcome of the project:

For both NAMD and NEMO it is common for users to run fewer processes than physical CPU cores due to memory limitations. For NAMD this would lead to poorer performance, but we can address this by running threads on the free cores. For NEMO there is little performance loss even in the absence of threading, and we can use the free cores to run the XIOS I/O servers. We also find that scalability is improved in NAMD by having some cores entirely dedicated to communication. In all cases it is important to match the distribution of processes and threads to the NUMA architecture of the nodes. These optimisations are based on the Tier-1 machine ARCHER2 with a many cores per node architecture such that they may be applicable to current and future Tier-0 architectures. Further performance tests and tuning based on PRACE Tier-0 Joliot-Curie is ongoing at the time of this report written. Although the processor and node architecture is very similar to ARCHER2, the network is completely different. We would expect to see the same qualitative effects in terms of performance, but the optimal parameter choices could be different. The impact on those two large communities' applications is important thanks to the work done, and we would expect to save millions of CPU hours on the targeted systems for those communities.

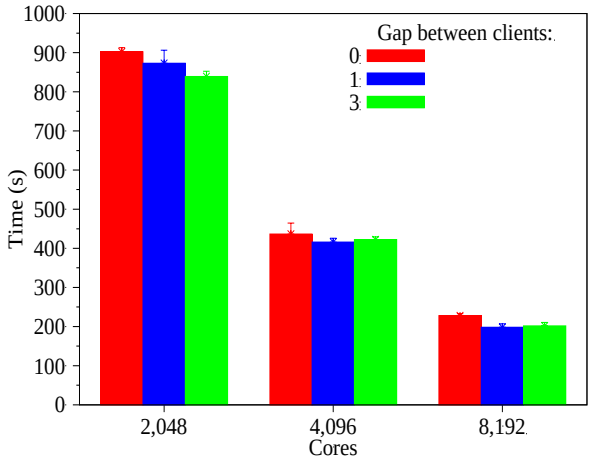


Figure 50: Investigating NEMO with the regular packing of ocean clients and idle cores, using 2 XIOS I/O servers per node and 4 cores per I/O server.

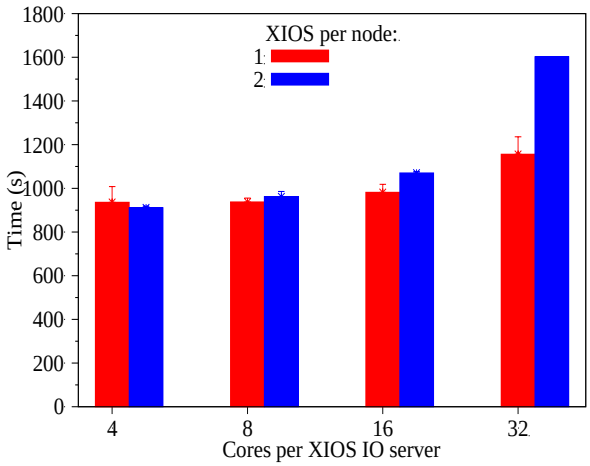


Figure 51: Varying the balance of XIOS I/O server cores per node on 2,048 cores. The remainder of available cores are populated with NEMO ocean

5.3.2 CaSToRC: Speeding up Full Wavefield Migration for Geo-Imaging

Introduction

Within Tasks 5 of PRACE 6IP Work Package 7 Computational Scientists from CaSToRC, CyI joined forces with Prof. Eric Verschuur and Dr. Mikhail Davydenko from the Delft University of Technology and member of the Delphi Consortium to trace and improve performance of a Geo-imaging application for PRACE Tier-0 systems. During the project we worked on the optimisation of a Full Wavefield Migration [38][39] application code for 2D and 3D domains of the Delphi Consortium.

FWM is an iterative wavefield migration algorithm that is based on one-way wave extrapolation, an explicit mechanism to model propagation between depth levels FWMod [37], and a linear optimisation algorithm, such as CG, for updating the subsurface reflectivity estimation. FWM uses primary and higher order scattering (multiples) to enhance the subsurface imaging process.

A typical seismic survey takes around 1.25 Mio core hours on an HPC system with Intel Xeon Skylake chips by using a dataset of 2.5 TB. The first profiling results showed that the scalability is hindered by communication overheads (blocking MPI routines), and strong scaling is breaking down if 32 nodes are exceeded. To reach higher resolutions and to efficiently utilise larger computational resources it is necessary to extend the scalability towards 100 nodes, thus enabling the application code to utilise PRACE Tier-0 systems efficiently.

The report consists of four parts. In the first part we present results from profiling experiments on 1 and 128 MPI processes, using the initial version of the code. This is to identify the most computationally expensive parts in the code, and to understand how communication overheads grow with MPI parallelism. In the second part, we discuss some of the changes we performed to enhance communication and computation efficiency. In the third section, we will give a brief outlook on the on-going effort to further accelerate the code's efficiency using Graphics Processing Units (GPUs).

Profiling

The first step towards enhancing the efficiency of the FWM application was to identify critical parts of the code, and to quantify the associated computations cost (in time) of each part. To do

so, we used the profiling tool Score-P/6.0 [41] , which allowed us to obtain insights to the computational kernels and communication routines by running with single or multiple MPI processes.

For the profiling experiments we used a problem with the following dimensionalities:

- Number of depth-steps: NZ = 361
- Number of sources: NS = 81
- Number of propagating frequencies: NF = 84
- Aperture dimensions: NX = NY = 21
- Wave-extrapolation algorithm: PSPI using max 6 ref. velocities
- Number of iterations: niters = 5
- Estimated memory footprint: 16.1 GB

Using 1 MPI process

The most time-expensive parts of the code are shown in the diagram of Figure 52. The figure shows the high-level as well as the lower-level functions, and the arrows indicate the invocation directions. As we can see, the main part of the code is given by the function `Slide`, which performs many important operations such as preparation of propagation operators, forward and backward in-time wavefield extrapolations (most expensive part), imaging at receiver's locations, and others. This part takes approximately 99.6% of the total runtime. The core of the extrapolation routine is given by the function `fft2d`. `ftt2d` performs forward and inverse FFTs on 2D signals, and consumes approximately 37% of the total runtime. Another expensive operation is a function called `getval`, which takes approximately 18.8% of total runtime. The self-time in functions `Slide` and extrapolation take 16.9% and 25.6% of the total runtime respectively. The MPI calls consume negligible time due all workload is carried out by 1 MPI process.

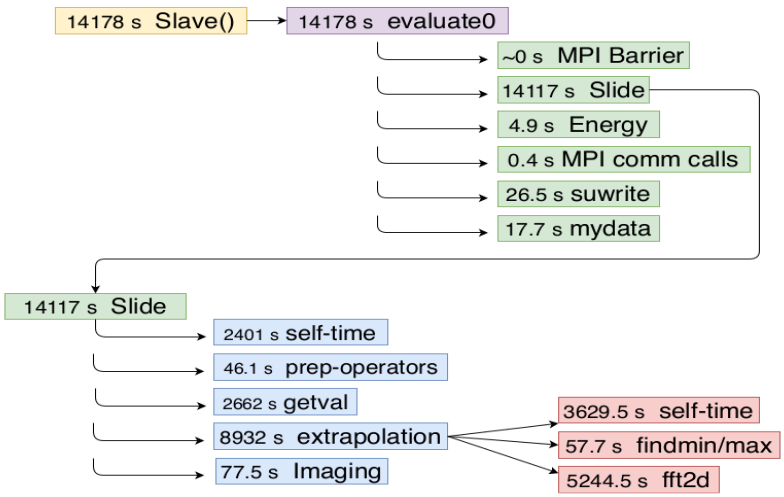


Figure 52: Profiling diagram of the FWM code using 1 MPI processes.

Using 128 MPI processes

To obtain an insight into the communication overheads, we profiled the application using 128 MPI processes. The profiling results are illustrated in Figure 53.

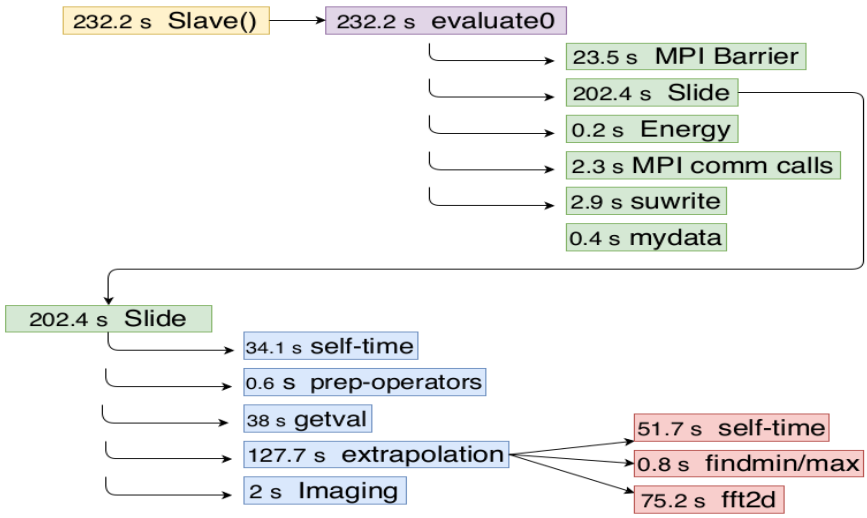


Figure 53: Profiling diagram of the FWM code using 128 MPI processes.

Similar to Figure 52, the dominant operation is `Slide`, which consumes approximately 202 out of 232 seconds. This corresponds to a percentage around 87.2% of the total runtime. Compared to the previous profiling experiment the percentage dropped from 99.6% to 87.2%, which is due to MPI overheads. Namely the parallelisable workload in each process becomes smaller, while the constant-time parts (i.e. I/O, memory allocations etc.) remain the same. The extrapolation routine takes 127.7 seconds, which is approximately 55% of total runtime, and the largest part of this operation is given again by `fft2d` that consumes 75.2s, or else, 32.4% of runtime. The `MPI_Barrier` calls take approximately 10.1% of the total runtime, and the MPI communication calls approximately 1%. All the communication and synchronization routines are called/invoked from the high-level function `evaluate0`. According to the diagram, most MPI-related overheads are synchronization (`MPI_Barrier`), rather than data movement (`MPI_Send/Recv`) overheads.

In addition to profiling the computations and communications using Score-P, we also performed strong scaling experiments on a local 8-node machine for three different problem sizes, which vary in memory and computation footprint by increase in the number of sources. The scaling results are shown in Figure 54. Experiment-size2 is four times larger than experiment-size1, and experiment-size3 is twice as large as experiment-size2.

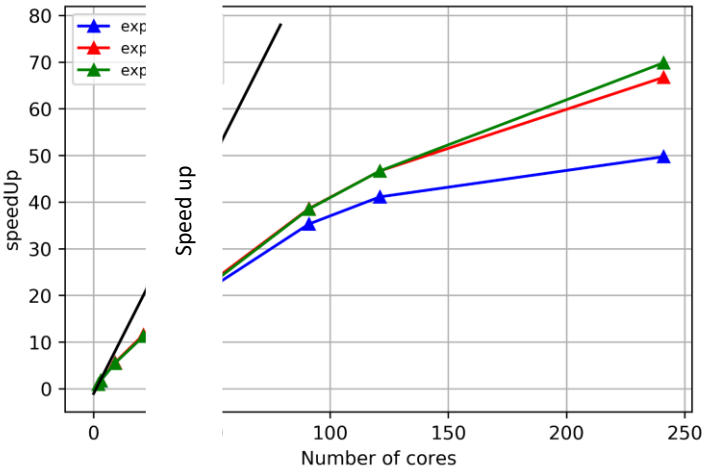


Figure 54: Strong scaling experiments of three different problem sizes that grow in the number of sources.

We observe that the scaling is limited, although the load-balance within the MPI parallelism is equally distributed, since each MPI process is processing exactly the same workload. The reason for the poor parallel efficiency of the code is due to global communications and other synchronisation overheads. Figure 55: Percentage from total runtime of MPI global communication and synchronisation overheads. shows how the global communication overheads (percentage from total runtime) grow with the number of MPI processes. Note that we ignored the peak, which appears when running with 9 MPI processes for the following interpretation and optimisation because it appears in the low parallelisation region.

To identify the cause for the MPI-overhead, we identified all MPI-calls within the source code. Apart from necessary communication routines, we observed an abundance of synchronisation barriers, placed in order to guarantee the correctness of the calculation. This increases the MPI overhead, causing as well difficult to explain variations in communication and synchronisation costs (e.g. the peaks at 9 MPI processes in Figure 55).

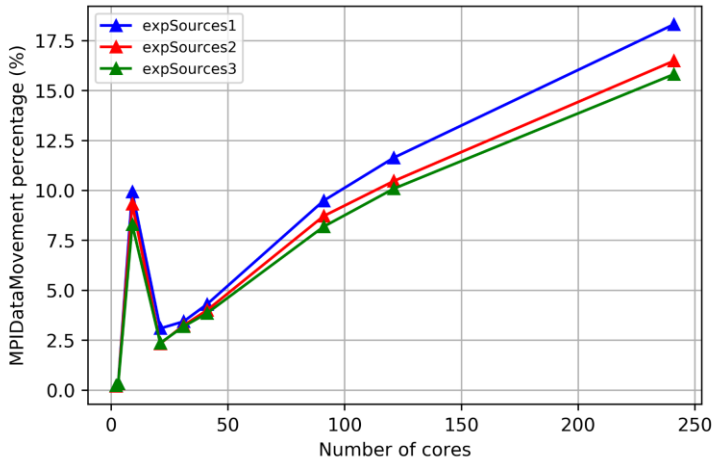


Figure 55: Percentage from total runtime of MPI global communication and synchronisation overheads.

In the following section we describe the implementation changes that were done in order to improve the efficiency of FWM code, and in extent discuss what other modifications could be attempted with the potential to further improve the application's runtime.

Improvements in Existing FWM code

The implementation changes in the existing FWM code focused on various improvements of computation and communication efficiency. To enhance computation's efficiency, the memory layout of the main data structure was optimised, and some source-code files were reorganised in order to help the compiler perform better optimisations and inlining. To enhance communication's efficiency (scalability) we used collective communication routines where possible, and removed unnecessary synchronisation barriers.

Code modifications

The main data structure in FWM is a 5-dimensional array implemented as a 5D pointer array `*****val`, accessible using `val[i][j][k][l][m]`. The initial implementation did not guarantee that data was stored in a contiguous block in heap memory, and this has some significant disadvantages with respect to performance. One disadvantage is that the non-contiguous data accesses result in significantly more cache misses, due to "jumps" in different positions in memory. Another disadvantage is that those data cannot be referenced, or copied efficiently for example when MPI communication is involved, resulting again in inefficient communication operations. We improved this by allocating data in contiguous memory chunks, whilst preserving the access mechanism (`val[i][j][k][l][m]`) too.

Moreover, according to the profiling, we observe that the function named `getval` consumed a significant amount of time, approximately 19%. This is quite a large overhead, considering that this function calculates an index using 5 integer parameters, and then returns the value stored in an array under this index. We found that the index function is extensively called, two orders of magnitude more times than any other function, however, the function and the calling-position were compiled separately, so the compiler was not able to inline this function and optimise the operation. In order to improve this operation, some parts of the source code were reorganised so that the compiler can inline this small function at the calling position and optimise the operation.

Furthermore, we analyzed the MPI communication within the function `MPI_Stack`, which was based on `MPI_Send/Recv` calls, and replaced it with `MPI_Allreduce`. This drastically reduced the MPI calls. Namely, instead of O(N) messages, like in the un-optimised code, `MPI_Allreduce` is based on log-tree communications, which require O(log(N)) number of messages. Thus, replacing point-to-point communication routines by collective MPI routines reduced the number of MPI-function calls and minimised communication overheads.

In the final step, we identified and removed all the unnecessary synchronisation calls, `MPI_Barrier` from the body of the function `evaluate0`, which is the main workload-function.

Improvement evaluation

After applying the developments that were described in the previous section, we re-evaluate the optimised application by re-running the same strong scaling cases, in order to compare and evaluate the improvements. The results are depicted in the scaling plots.

In Figure 56 we calculate the total runtime improvements for experiments with varying numbers of MPI processes using the formula:

$$\% \text{ of improvement} = \text{time\_before} - \text{time\_after} / \text{time\_before}$$

We observed that the runtime improvement is increasing with the number of MPI processes and this is associated with the global reduction optimisation as well as the removal of synchronisation barriers. We also observe a 5-10% improvement in runtime on fewer number of processes, which possibly is due to better functions inlining, and the improved memory layout of the main data structure. On larger numbers of MPI processes, where contribution comes from all reported code modifications the improvement is even greater.

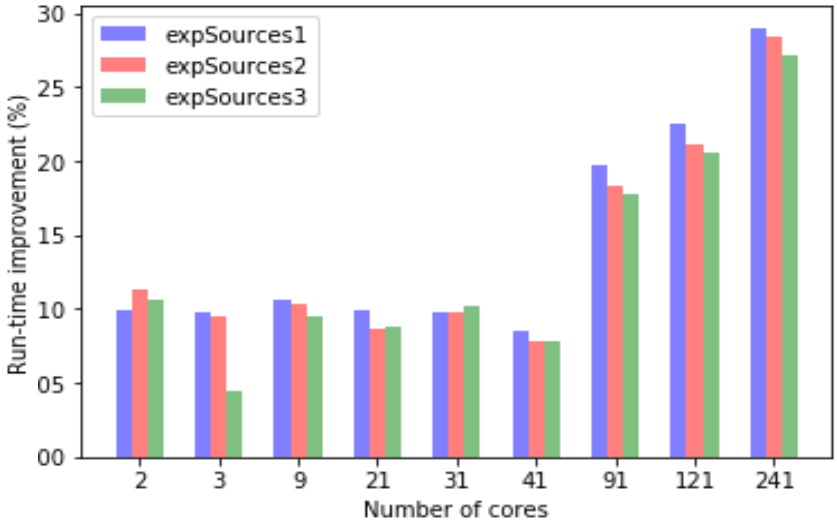


Figure 56: Percentage of runtime improvement compared to the initial version of the code.

In addition, we demonstrate the new strong scaling plot of the application in Figure 57. Since we used the same problem examples as the ones used in the profiling section, we can make a direct comparison with the strong scaling of the application before applying the code modifications from Figure 54. The code modifications had a positive impact on the scalability of the applications. We observe that the maximum scaling for the largest problem is now above 80, whereas before was around 70 on 241 MPI processes. For the smallest problem we have scalability improvement as well, from speedup 50 on 241 processes to above 60. The curve for the middle-size problem lies between the two, both in Figure 54 and Figure 57.

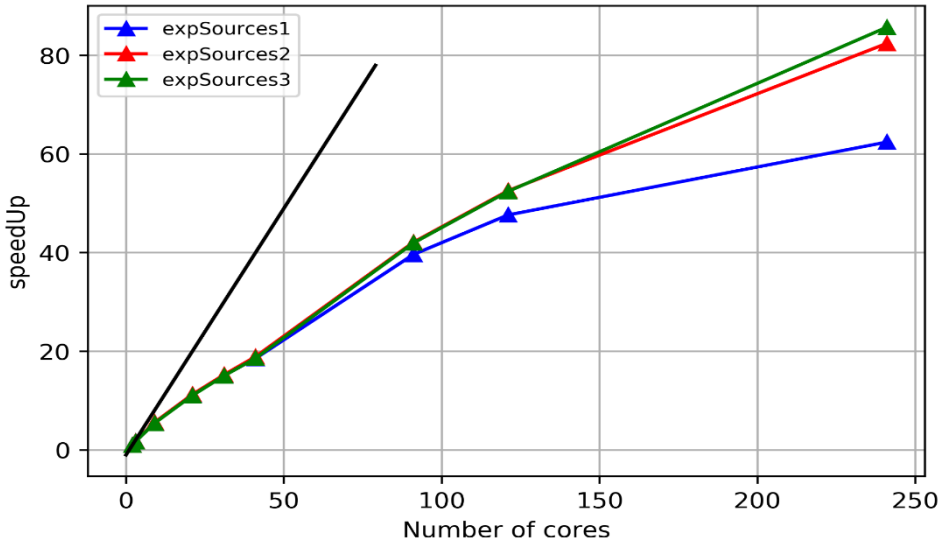


Figure 57: Strong scaling experiments of three different problem sizes that grow in the number of sources, using the improved code.

Large scaling experiments

To evaluate the performance of the FWM application on a PRACE Tier-0 system, we selected JUWELS Cluster, a HPC system equipped with around 2000 compute nodes, equipped with Intel Xeon Skylake chips (each compute node is equipped with 2x Intel Xeon Platinum 8168 CPU). JUWELS Cluster is hosted by the Juelich Supercomputing Center and is part of the PRACE Tier-0 systems.

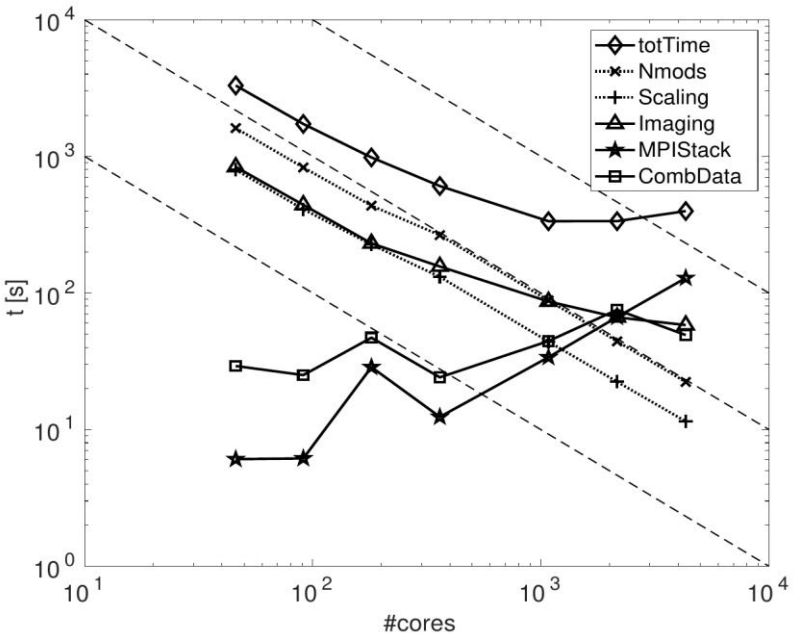


Figure 58: Scalability of the different parts within an iteration of the FWM application without improvements.

Figure 58 shows the scalability of the different parts within an iteration of the FWM application without improvements. It is obvious that the communication routines `MPIStack` and `CombData` do not scale.

We performed a similar performance test by scaling up to 90 nodes. To identify bottlenecks and evaluate performance improvements, we timed the major communication and computing routines. We benchmarked one iteration of the FWM process employing 81 sources and 160 frequencies. With the obtained numbers, we can estimate the computational cost of a production run, typically consisting of O(1000) sources, O(100) different frequencies and O(40) iteration within the imaging process.

We list the obtained numbers in Table 13 and depict the scalability in Figure 58 for old applications. The time needed for the communication routines `MPIStack` and `CombData` increases with a larger number of processes due to its Point-to-Point communication structure. Replacing the Point-to-point routines with collective communication routines in `MPIStack` resolves that bottleneck, leading to a decrease of the communication time for larger number of nodes, as listed in Table 14. A possible explanation could be that the communication is bandwidth bounded, which minimises the communication time if communicating less data to a larger number of nodes with more available communication channels.

np	Modeling (s)	Scaling (s)	Imaging (s)	MPIStack (s)	CombData (s)
46	1609.89	797.31	836.83	06.07	29.2
91	829.13	409.96	441.32	6.15	25.05
181	435.46	226.10	230.6	28.66	47.05
361	265	131.17	156.18	12.38	24.11
1081	88.05	44.03	86.62	33.85	44.44
2161	44.23	22.37	66.31	66.62	75.27
4321	22.3	11.47	58.21	128.64	49.17

Table 13: Timing of the different parts of an imaging iteration within the FWM application in its initial phase without improvements.

np	Modeling (s)	Scaling (s)	Imaging (s)	MPIStack (s)	CombData (s)
46	1531.57	759.71	794.13	14.23	34.28
91	791.64	392.5	421.01	9.58	30.36
181	414.57	215.5	219.17	23.67	41.3
361	253.93	128.11	151.93	3.15	26.67
1081	84.84	42.35	85.29	0.82	46.19
2161	42.65	21.36	61.98	0.72	41.2

np	Modeling (s)	Scaling (s)	Imaging (s)	MPIStack (s)	CombData (s)
4321	21.47	10.95	55.26	0.46	46.85

Table 14: Timing of the different parts of an imaging iteration within the FWM application with improvements of computational and communication kernels.

In total the total computational cost could be reduced by a factor two if the largest tested parallelisation is compared, see also Figure 59. Namely 40 imaging iterations on 4321 cores costs around 13 000 core hours with the initial applications, while for the application optimised within PRACE 6IP WP7T5 only 6560 Core hours are needed. Note that within a production run it a reduction to 2161 cores would also reduce the total cost per application to 6661 core hours for the old and 4070 core hours for the optimised version. However, that also increases the run time from 1.5 hours to 1.9 hours for the new version while the initial application does not scale and requires 3 hours on 4321 cores and on 2161 cores. Note that in case of a production run, the imaging process can be sped up by a factor 2 utilising the same computational costs. Moreover utilising 50 nodes would lead even to less computational need with a speed up by a factor 1.5.

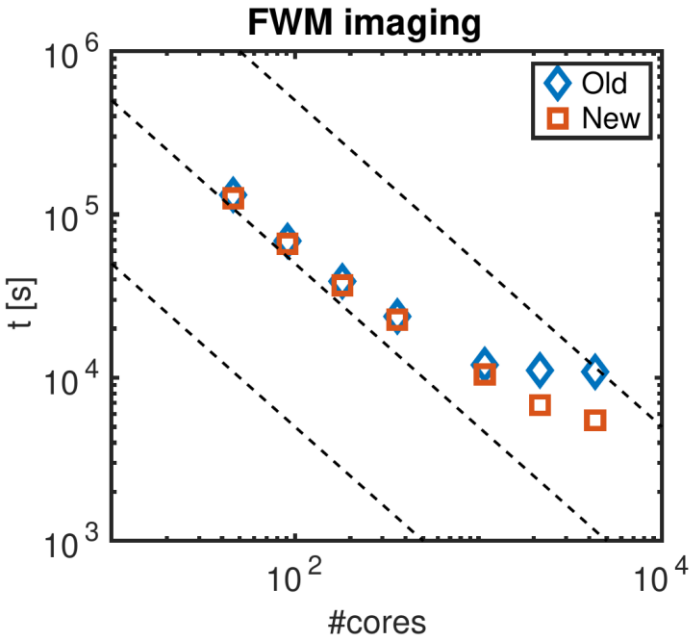


Figure 59: Comparison of strong scalability of the initial FWM application compared to the improved application on JUWELS Cluster for one imaging process consisting of 40 iterations.

Additional work

In addition to the improvements that were applied directly to the FWM code, this work allowed us to identify the computational hot-spots of the application. An important component of the algorithm is given by the propagation of wavefields between depth levels, which can be done via phase-shift plus interpolations.

The phase-shift plus interpolation algorithm [36][40] was investigated separately through profilings of two and three dimensional wavefield propagation test-case applications, which were developed and optimised to achieve high efficiency. In these developments special attention was given to the layout of the data in memory, especially to evaluate the best practice

to achieve cache friendly operations, limit the data copies, avoid unnecessary memory reallocations and frees, and effective usage of optimised libraries in critical regions such as Fourier Transformations.

This led to an extension of our developments to multi-threaded CPU implementations using OpenMP, as well as GPU implementations using the CUDA and HIP programming interfaces. We successfully ported the test-case applications to NVIDIA P100, NVIDIA V100, and AMD Radeon Instinct MI100 GPUs. Preliminary performance experiments indicate computations speed-up by a factor 5 on CPUs, and by a factor 25 on GPUs are possible due to the investigations for the test-case applications, relatively to the same baseline. Further efforts are here needed to reach production level, which are on-going but beyond the tasks of WP7 of PRACE 6IP.

Outcome of the project
Within the project we could successfully identify bottlenecks of the Full Wavefield Migration application by using Score-P and strong scaling tests. The optimisation led to an improvement of the scalability and of the computation of the code. This enabled on-going work to port the computational kernels to GPUs, which potentially will further improve and speed up the computational kernels. Note that the improvements within PRACE 6IP WP7 task 5 were timely to be utilised within the Center of Excellence RAISE, where the FWM code is one of the applications targeted for acceleration via machine learning for the European exascale era.

5.3.3 IT4I

For our contribution to the HLST project we picked the two research groups with the highest CPU.hour usage of our system.

The first is the material design research group lead by Dominik Legut. This group is part of Modeling for Nanotechnologies Lab at IT4Innovations. Two specific cases have been addressed for this research groupe: ReaxFF simulations on GPU using LAMMPS with Kokkos package, and VASP calculations on GPU using version 6.2.1 with OpenACC support.

The second with highest core hour consumption is a computational chemistry research group lead by Pavel Hobza. This research group is focused on computer-aided drug design, but the people involved in this project work in material design. These people are affiliated at Modeling for Nanotechnologies Lab at IT4Innovations, but they are also a part of the Czech Academy of Sciences. Two test cases have been addressed for this research group: CCSD(T) calculation in Molpro and again VASP 6 calculations on GPU.

ReaxFF simulation using LAMMPS with the Kokkos package
The advance of computing power has opened new perspectives of computational materials design. Accurate predictions of the intrinsic physical properties are routinely obtained by performing ab-initio simulations at the atomistic level. These calculations, however, have severe restrictions on the size and temporal evolution of the systems. Finite-temperature properties of nano- and meso-scale systems (10^4 to 10^7 atoms) are simulated via classical molecular dynamics. The atomic interactions are described, however, by parametrised force-fields, which lack the transferability and desired accuracy over different temperature and pressure conditions. A reactive force-field potential, ReaxFF, offers a trade-off between accuracy and system size. Unfortunately, the existing implementation of the ReaxFF package of widely used LAMMPS code does not offer a straightforward parallel optimisation on new

hybrid CPU-GPU architectures. Within this project, we aim to use the accelerated options of the LAMMPS code for efficient parallel execution of the ReaxFF package on a GPU architecture.

The LAMMPS implementation offers the possibility of parallel execution of ReaxFF on GPUs through KOKKOS libraries. Our initial testing on IT4I's Barbora system with a 36-core CPU node with Intel Cascade Lake CPUs compared to a single GPU node with 4 V100 GPUs showed approximately 14x speedup in favour of the GPU node when performed on a CPU node vs a GPU node. A 1 ns long simulation with 1 million atoms on 4 CPU compute nodes would take about 90 days on Barbora. Our goal for this project would be to run this simulation in a week.

After the initial testing we applied for Preparatory Access 2010PA5586 for the JUWELS Booster system. We used the LAMMPS stable release from 29 October 2020, compiled with CUDA-Aware MPI. Modules used for compilation were: GCC 10.3.0, CUDA 11.3, and OpenMPI 4.1.1. The application was compiled using CMake and make build systems. The LAMMPS packages included in the compilation were: Body, Kspace, Manybody, Phonon, Qeq, Reaxff, and Kokkos. The input data were taken from the LAMMPS website with benchmarks. (`bench_reaxc.tar.gz` from https://www.lammps.org/bench.html) The results of the scaling measurements are shown in Table 15.

Number of atoms	1 node 4 GPUs	2 nodes 8 GPUs	4 nodes 16 GPUs	8 nodes 32 GPUs	16 nodes 64 GPUs	32 nodes 128 GPUs	64 nodes 256 GPUs
0.25 million	41.11 s	36.19 s	26.11 s	23.48 s	23.52 s	24.85 s	26.79 s
	1.00 ×	1.14 ×	1.57 ×	1.75 ×	1.75 ×	1.65 ×	1.53 ×
0.5 million	65.01 s	50.46 s	32.68 s	26.20 s	25.57 s	26.32 s	27.51 s
	1.00 ×	1.29 ×	1.99 ×	2.48 ×	2.54 ×	2.47 ×	2.36 ×
1 million	110.04 s	79.62 s	45.89 s	33.47 s	28.35 s	28.08 s	29.30 s
	1.00 ×	1.38 ×	2.40 ×	3.29 ×	3.88 ×	3.92 ×	3.76 ×
2 million	198.95 s	134.98 s	70.54 s	46.72 s	36.11 s	31.21 s	31.13 s
	1.00 ×	1.47 ×	2.82 ×	4.26 ×	5.51 ×	6.37 ×	6.39 ×
4 million	375.88 s	242.63 s	115.85 s	71.73 s	49.82 s	39.25 s	34.77 s
	1.00×	1.55 ×	3.24 ×	5.24 ×	7.54 ×	9.58 ×	10.81 ×
8 million		456.31 s	206.88 s	118.42 s	84.26 s	55.12 s	43.46 s
		2.00 ×	4.41 ×	7.71 ×	10.83 ×	16.56 ×	21.00 ×
16 million			385.58 s	209.28 s	133.85 s	83.47 s	61.05 s
			4.00 ×	7.37 ×	11.52 ×	18.48 ×	25.26 ×

Table 15: Runtime and achieved speedup compared to the calculation on a single GPU node for 1000 simulation steps with 0.1 fs timestep. The scaling was measured with the number of atoms ranging from 0.25 million to 16 million. Test case with 8 million atoms did

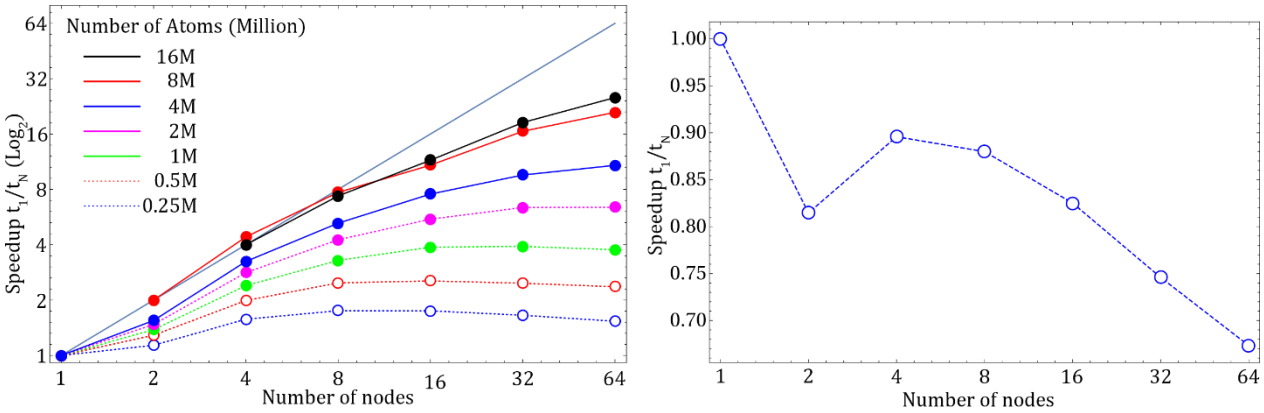


Figure 60: The data from Table 15 are shown in this figure. The left plot displays the strong scaling for workloads with different number of atoms. Both axes are in base 2 logarithm. The right plot shows the weak scaling with its parallel efficiency when each node calculates 0.25 million atoms.

The strong scaling shows that the efficiency of parallel execution on GPUs when using ReaxFF increases with the size of the system. We assume that the poor performance of smaller system sizes is caused by a communication bottleneck. Thus, for the best performance, one should consider running calculations for systems with up to 100 of millions of atoms. For a system this large based on the weak scaling plot we can estimate a parallel efficiency above 50 %.

To make the scaling somewhat efficient on Tier-0 GPU accelerated systems an optimal system size should be in order of 100's millions of atoms. The material design group did not have a use case for calculations of such large problems at the moment, so we did not proceed to apply for Project Access with this type of calculation. Our original goal was to determine whether we are able to calculate a 1 ns simulation of 1 million atoms in less than a week. On 8 GPU accelerated nodes of JUWELS Booster this would take less than 4 days.

VASP 6.2.1 calculations on GPU with OpenACC directives

VASP is one of the most versatile and stable plane-wave codes, with the accuracy of all-electron methods. Most importantly, it provides an accurate set of potentials, making it the software of choice for the computation of materials properties. Another advantage of the code is the robust structure optimisation and accurate forces. And finally, its latest version with OpenACC allows to run most of the code features on multiple GPUs efficiently.

There was some support for GPU calculations using CUDA in earlier versions of VASP. However, it lacked support for reciprocal space projection, which made the calculation less accurate. In version 6 of VASP the support for reciprocal space projection on GPU was added using OpenACC directives.

This research aims to clarify the structural and physical properties of advanced 2D van der Waals (vdW) materials using the state-of-the-art computational materials research methods, towards the development of energy-harvesting materials and next-generation electronic devices. The 2D vdW materials offer great advantages to design heterostructures of superior properties through band structure and phonon spectrum engineering. However, due to the corrugation, strain distribution, and the so-called Moire structure in the 2D materials, one needs to treat very large atomic systems, which is difficult to tackle using standard non-accelerated CPUs at HPC clusters. The project focuses on prediction of the novel 2D vdW-based thermoelectric materials.

In this project we initially compiled VASP with OpenACC support on IT4I system Barbora and later Karolina. The compilation was done from sources using makefile build system. For each

system the makefile.include configuration file was created by providing correct paths for all the used modules and libraries. The compilation was done by nvfortran compiler from Software Development Kit from Nvidia (NVHPC SDK). Mathematical libraries: BLAS, LAPACK and ScaLAPACK were also provided by this SDK. For parallelisation an OpenMPI with CUDA-Aware support was used.

After doing initial tests and verifying the results from the compiled GPU version we used our Preparatory Access 2010PA5586 to JUWELS Booster from the previous LAMMPS project to measure the scaling. It was necessary to adjust the makefile.include configuration for the environment available on the Booster system. NVHPC SDK was used for compilation, mathematical libraries were used from Intel MKL and ParaStation MPI supporting CUDA-Aware features was used for parallelisation.

After having the compilation ready we ran a series of tests with a production jobs to determine scalability. Explanation of simulation parameters:

- NKPTS – number of k-points in the irreducible Brillouin zone (determined by KSPACING parameter, size of the simulation cell and symmetry)
- NBANDS – number of electronic bands (determined by total number of electrons / sort and number of atomic species)
- ENCUT – cut-off energy which determines maximum number of plane-waves

These parameters determine the required memory and if the system can be calculated on one single node. In addition, the value of NKPTS determines the number of nodes the problem can be efficiently scaled.

Test case 1:
SYSTEM=Co_pvFe_pvTa_pv, (CoxFe1-x)2Ta hP24 phase, x=0.25;
2×2×1 supercell: 16 Co, 48 Fe and 32 Ta atoms;
PREC=Accurate, ENCUT=527.828(eV) (1.8×ENMAX), ISMEAR=1, SIGMA=0.10,
KSPACING=0.10, NSIM=16;
NKPTS=30, NBANDS=960, maximum number of plane-waves: 34296

N_{nodes}	1	2	3	5	6	10	15	30
t_{LOOP} (s)	107.75	54.11	36.09	21.90	18.22	13.31	9.00	3.93
Mem./node (%)	8.4	8.4	8.4	8.4	8.4	8.4	8.4	8.4
Mem./GPU (%)	34	34	34	32	33	32	32	32

Table 16: Average time of one electronic iteration, tLOOP, during the first 10 electronic steps. Parameter KPAR=Nnodes.

Test case 2:
2×2×2 supercell: 32 Co, 96 Fe and 64 Ta atoms
KSPACING=0.10
NKPTS=20, NBANDS=1916; maximum number of plane-waves: 68552

N_{nodes}	1	2	4	5	10	20
t_{LOOP} (s)	382.69	191.90	78.34	62.94	31.81	16.34

N_{nodes}	1	2	4	5	10	20
Mem./node (%)	18.8	18.8	18.8	18.8	18.8	18.8
Mem./GPU (%)	81.4	76.5	74.1	74.1	69.4	69.4

Table 17: Average time of one electronic iteration, tLOOP, during the first 10 electronic steps. Parameter KPAR=Nnodes.

Test case 3:
3×3×2 supercell: 72 Co, 216 Fe and 144 Ta atoms;
KSPACING=0.20, NSIM=16;
NKPTS=6, NBANDS=4000; maximum number of plane-waves: 154170

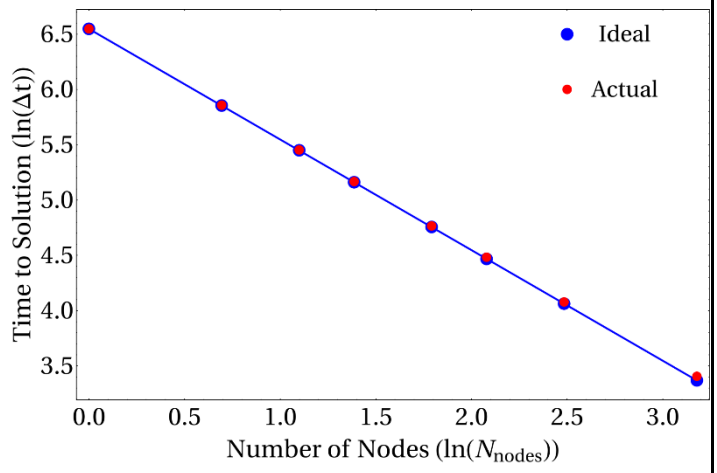
N_{nodes}	1	2	3	6
t_{LOOP} (s)	776.35	389.83	261.25	130.88
Mem./node (%)	26.4	26.4	26.4	26.4
Mem./GPU (%)	99.7	99.7	99.7	99.7

Table 18: Average time of one electronic iteration, tLOOP, during the first 10 electronic steps. Parameter KPAR=Nnodes.

Test case 4:
SYSTEM=Fe_pvTi_pv (Fe2Ti Laves phase, 3×3×3 supercell: 216 Fe and 108 Ti atoms),
PREC=Accurate, ENCUT=293.238(eV) (1.×ENMAX), ISMEAR=1, SIGMA=0.10,
KSPACING=0.08, NSIM=16;
NKPTS=24, NBANDS=3008; maximum number of plane-waves: 47110

N_{nodes}	1	2	3	4	6	8	12	24
t_{LOOP} (s)	697.41	349.60	233.41	175.39	117.23	88.51	59.11	30.12
Mem./node (%)	25	25	25	25	25	25	25	25
Mem./GPU (%)	94	94	94	94	91	91	91	89

Table 19: Average time of one electronic iteration, tLOOP, during the first 10 electronic steps. Parameter KPAR=Nnodes.



Nodes	Time to Solution (s)	Ideal Time to Solution (s)	Parallel Effic. (%)
1	697.41	697.41	100
2	349.60	348.70	99.7
3	233.41	232.47	99.6
4	175.39	174.35	99.4
6	117.23	116.23	99.2
8	88.51	87.18	98.5
12	59.11	58.12	98.3
24	30.12	29.06	96.5

Figure 61: For this test case we include a plot showing the parallel efficiency from 1 to 24 GPU nodes. When we run this test case on 24 GPU nodes the parallel efficiency is 96.5%.

Based on these scaling tests we determined a few conditions for running the OpenACC port of VASP on JUWELS Booster. The code can be parallelised up to N_{nodes} = NKPTS if KPAR is set to the number of nodes or up to N_{nodes} / n where KPAR = n * N_{nodes}. For this architecture an optimal workload would be a system of NBANDS = 2000 – 3000, with approximately 50000 plane waves and number of k-points = 24 – 32.

Splitting a single k-point over multiple GPU accelerated compute nodes does not have a particular speedup benefit. It does give a benefit of splitting the memory in-between these nodes so this would allow for running larger workloads that do not fit into the GPU memory of a single node. But with our testing the simulations which were run with these settings did not run properly. The simulation got stuck and froze during a random electronic step. We did not find the cause of this issue. This issue was present in all the compilation settings we used on all systems that we tried.

Based on these data we applied for 23rd Call of PRACE Project Access with project: Novel 2D thermoelectric materials. Unfortunately, this project was not approved. Nevertheless, the research group started using this GPU version of VASP on our new local system Karolina with 72 GPU accelerated nodes. Each node contains 8 A100 GPUs, therefore more GPU computing power and more GPU memory on a single node than on JUWELS Booster.

In the period of 2019-2020, 2/3 of all core hours spend in IT4I were allocated to projects related to material sciences. And we estimate that 1/3 of that allocation was used by VASP users. That can translate up to 150 million of core-hours spend on VASP calculations. For the suitable class of problems the parallel efficiency of the GPU version of VASP consumes at least 5 time less core hours (assuming that there is no extra coefficient for running on GPU node).

We also presented these findings and experiences with the GPU version of VASP on the 5th Users' Conference of IT4Innovations held on November 9, 2021. So that more VASP users can start utilising GPU accelerated nodes and run their calculations more efficiently. The GPU version of VASP will be made available as a module for all holders of the VASP 6 license.

CCSD(T) calculation with Molpro

In periodic models, it is possible to perform mostly DFT-based calculations. If something more is needed to be done, a periodic model must be simplified to the cluster. The cluster model calculations of surfaces are often performed, which are not available in periodic models calculations with same accuracy. There are a few limits of the cluster model, like boundary conditions, which means that the cluster model must be chosen with taking special care of these limits.

Especially, cluster models of metal surfaces contain a lot of heavy elements, which is in itself problematic for some post-Hartree-Fock methods like MP2 or CCSD(T). On our cluster models we need to perform single-reference calculations, but also multi-reference calculations e.g. MRCI or CASPT2. Both types can be effectively calculated by the program Molpro. We made a model system and used that Molpro tests. Our model contains 24 atoms and was calculated in the heavy-aug-cc-pVTZ basis set. In the future, we will have larger models, but for initial testing of Molpro, this should be enough.

These types of calculations were run on IT4I's latest system Karolina. The compute node consists of two 64-core AMD CPUs and 256 GB of memory. For this type of calculation, the user was not setting any parallelisation parameters (number of MPI processes, OpenMP threads). The default setting meant that the calculation was running in 1 MPI process with 128 OpenMP threads. This resulted in a very poor performance when the calculation of a single complex was running for days. This type of calculation is also very memory demanding, so the jobs were often crashing by the lack of memory. Molpro is also very I/O intensive and during computation it stores large temporary files on the disk and, also reads them very frequently.

For the performance measurement we ran a series of tests with the same workload to determine the best configuration for this type of calculation. Different numbers of MPI processes and OpenMP threads were used. The test was run on the regular compute node with 256 GB of memory as well as on GPU accelerated node which has 1024 GB of memory. The usage of this increased memory was tested as a RAM disk instead of the standard SCRATCH storage. We also tried to spread the calculation over multiple nodes to see if it can give any speedup. For our testing we used Molpro installed on the cluster from binaries. We did not compile our own version of Molpro since it has many dependencies, and it would be too difficult to setup the compilation configuration.

Available memory (GB)	Disk type	Compute nodes	MPI procs per node	OpenMP threads	Memory per proc (Mwords)	Global Array size (Mwords)	Simulation runtime (h)
1024	scratch	2	16	8	5500	30000	6.16
256	scratch	2	24	5	1270	300	6.90
1024	scratch	2	8	16	11000	30000	7.14
1024	scratch	1	24	5	5066	300	7.50
256	scratch	2	16	8	1300	8500	7.53
256	scratch	1	24	5	1270	300	7.63
1024	ramdisk	1	16	8	1300	8000	8.15
1024	scratch	1	16	8	6000	25000	8.17
1024	scratch	1	16	8	7600	300	8.35
1024	scratch	1	32	4	3800	300	8.40
256	scratch	1	16	8	1300	8500	9.55
1024	scratch	1	8	16	15200	300	10.40
1024	ramdisk	1	8	16	2500	8500	10.47
1024	scratch	1	8	16	11000	30000	10.85
256	scratch	1	8	16	2500	9000	14.16

Table 20: Comparison of different Molpro settings for the same calculation and its influence on the runtime.

The parameter with the most influence on the performance is the number of MPI processes and correspondingly also the number of OpenMP threads. The ideal amount for this calculation is 24 MPI processes on a single node. The influence of ramdisk used as a scratch disk is not as significant as we expected. The scratch file system can provide enough bandwidth for the application needs. The influence of ramdisk can be seen only during the initialisation phase during SCF calculation. But overall, if we compare the same configurations run with ramdisk and scratch there is not much difference in the total execution time between those two.

Running the calculation on multiple compute nodes does not give adequate speedup compared to the additional computational time spend. This might be because only a certain part of the calculation can utilise the power of multiple nodes. Therefore, it is not efficient to run this type of calculation on multiple compute nodes.

The main scope of our work was to find the most optimal parameters of calculation. We reduced the execution time from days on multiple nodes to hours on a single compute node. Actually, we do not have enough computational jobs for utilisation of Tier-0 system. Instead of participating in PRACE Project Access we decided to apply for EuroHPC resources, namely the large memory partition of MeluXina cluster in Luxembourg. For even more memory demanding Molpro workloads the fat node is available on Karolina with 24 TB of memory.

Second VASP 6.2.1 calculations on GPU

Similarly, to the previous project with the OpenACC port of VASP 6 calculations done for the material design group, this one was done in the same way for the computational chemistry group. We used a lot of experience from the previous project. The advantage was that we already had a running version of VASP compiled on JUWELS Booster system. We had to apply for a new Preparatory Access to JUWELS Booster system since the previous one expired. We measured a new scaling data for different test cases provided by the computational chemistry group. From these data a PRACE Project Access Application was prepared.

Photoelectrochemical (PEC) water splitting is an encouraging approach for solar-driven hydrogen production through zero emissions, and it offers a promising strategy to convert and store solar energy in chemical bonds. TiO_2 has been employed as a photocatalyst since the

1970s because of its low cost, abundance, and stability. TiO_2 exhibits in several different polymorphs that all behave differently. The most used polymorphs, rutile and anatase TiO_2 are utilised for photocatalytic water splitting. However, the large band-gap in TiO_2 (3.0-3.2 eV) precludes their functioning in the energy spectrum corresponding to the visible light. In view of the overpotential requirements (0.4−0.6 eV) and energy losses (0.3−0.4 eV) during PEC water splitting, an ideal bandgap should be ∼2.0 eV, which corresponds to a light absorption edge of ∼620 nm. In recent years, several studies have been done aiming to surmount this obstacle by modifying the surface or depositing noble metals. Thus, high solar-to-hydrogen conversion efficiencies can be attained with proper bandgap energy, band edge positions and efficient charge separation during the PEC reactions.

In this project we have calculated the bandgap of TiO_2 with various facets such as (001), (101), and (111). Molecular dynamic calculations were also performed to measure the stability of the TiO_2 surface. In this case, we used Preparatory Access 2010PA6144 for JUWELS Booster to determine the scaling. We have performed a series of calculations of production size jobs. We have chosen the jobs, where there is requirement of large number of compute nodes while running VASP on CPU.

Test case 1:
SYSTEM= anatase TiO_2 (001) facets;
36 Ti, and 72 O atoms;
Method: GGA+U (U=8.0 eV)
PREC=Normal, ENCUT=400 eV, ISMEAR=0, SIGMA=0.05,
NSIM=1;
NKPTS=120, NBANDS=600, maximum number of plane-waves: 451584

N_{nodes}	1	2	3	4	6	10	15	30	40	60	120
t_{LOOP} (s)	293.90	147.15	98.20	73.75	49.12	29.53	19.75	9.94	7.51	5.05	2.59

Table 21: Average time of one electronic iteration, tLOOP, during the electronic steps. Parameter KPAR=Nnodes.

Test case 2:
SYSTEM= anatase TiO_2 (101) facets;
24 Ti, and 48 O atoms;
Method: GGA+U (U=8.0 eV)
PREC=Normal, ENCUT=400 eV, ISMEAR=0, SIGMA=0.05,
NSIM=1;
NKPTS=100, NBANDS=600, maximum number of plane-waves: 453600

N_{nodes}	1	2	4	5	20	25	50	100
t_{LOOP} (s)	109.26	54.76	27.43	22.58	5.56	4.47	2.29	1.20

Table 22: Average time of one electronic iteration, tLOOP, during the electronic steps. Parameter KPAR=Nnodes.

Test case 3:
SYSTEM= anatase TiO_2 (111) facets;
64 Ti, and 128 O atoms;
Method: GGA+U (U=8.0 eV)
PREC=Normal, ENCUT=400 eV, ISMEAR=0, SIGMA=0.05,

NSIM=1;
NKPTS=70, NBANDS=720

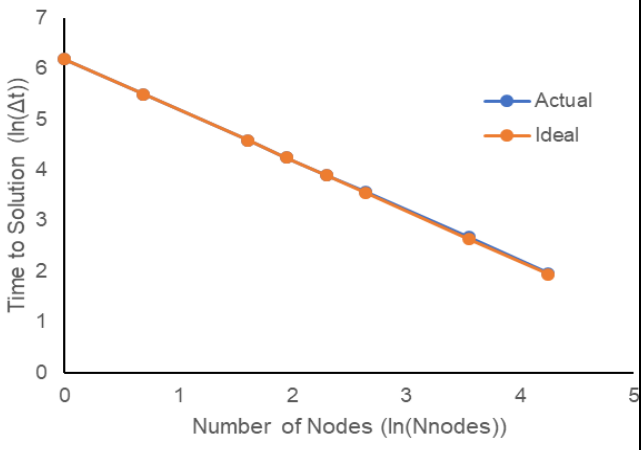
N_{nodes}	1	2	5	7	10	14	35	70
t_{LOOP} (s)	488.04	244.42	97.86	69.85	48.99	35.40	14.41	7.09

Table 23: Average time of one electronic iteration, tLOOP, during the electronic steps. Parameter KPAR=Nnodes.

Test case 4:
SYSTEM= anatase TiO_2.
36 Ti, 81 O, and 18 H atoms;
PREC=Normal, ENCUT=520 eV, ISMEAR=0, SIGMA=0.1
NSIM=1;
NKPTS=1, NBANDS=392, maximum number of plane-waves: 688128

fs (s)	10	20	30	40	50	60	70	80	90	100
t_{LOOP} (s)	20.36	17.86	24.55	35.37	25.06	25.21	25.99	24.41	25.03	26.54

Table 24: Average time of each 10 fs step, tLOOP, during the electronic steps. Parameter KPAR=Nnodes



Nodes	Time to Solution (s)	Ideal Time to Solution(s)	Parallel Effic. (%)
1	488.04	488.04	100
2	244.42	244.02	100
5	97.86	97.608	100
7	69.85	69.72	100
10	48.99	48.804	100
14	35.4	34.86	98.5
35	14.41	13.944	96.8
70	7.09	6.972	98.3

Figure 62: Plot of strong scaling from test case 3 with parallel efficiency.

We can see that the parallel efficiency when running on 70 GPU nodes is 98.3%. Although the bandgap calculation jobs are large in terms of memory, they were able to fit into the GPU memory on a single node. Since they have large number of k-points they can be well parallelised and run on 70 – 120 GPU accelerated compute nodes.

Initially, the bandgap calculations were run on IT4I's old system Salomon with 25 – 30 compute nodes. Each node consists of 24 compute cores of Intel Haswell architecture. This system will be decommissioned this year after 7 years of service. For test case 1, we have used total 600 cores and the time required to perform one electronic iteration, on average about 8725 seconds. Similar number of cores used for test case 2 and the necessary time for one electronic iteration is 7726 seconds. However, test case 3 contains large no. of atoms and we have used 720 cores for this calculation. The time required to calculate one electronic iteration for test case 3 on average about 8567 seconds. Although these data are from deprecated CPU architecture, they can give some idea about the efficiency improvement when these calculations are run on the

current generation of Nvidia GPUs. Therefore, we have applied for the 24th call of PRACE Project Access for 22 million cores-hours on JUWELS Booster system.

5.3.4 SNIC-UU: Free-energy perturbation calculations of protein-ligand binding affinities

In the present project, a multi-scale workflow for studying protein–ligand binding affinities is being ported to the Tier-0 system at the Jülich Supercomputing Center (JSC), more specifically the JUWELS Booster (JB) system. Calculating binding free energy of drug candidates to their receptor proteins is one of the greatest challenges in drug development: If the binding affinity could be accurately predicted, chemical synthesis of most of the drug candidates could be avoided, which would save an enormous amount of money and time. Moreover, this would be the ultimate form of green chemistry –no chemicals would be needed at all. The workflow is based on free-energy perturbations, which is a strict statistical-mechanics theory that gives the correct results provided that the potential-energy function and the sampling is extensive. The long-term goal is to improve existing methods with the use of QM calculations. To that aim, we have designed a large-scale test case of ~100 ligands binding to 10 proteins. We then need an effective method to calculate the free energies fast and effectively, using different improvements of the molecular-mechanics potential.

The present workflow is part of the research of our local user, Professor Ulf Ryde. This workflow is computationally heavy as it involves many independent simulations in order to get accurate ligand binding free energies. Although our local cluster (LC) offers GPU support in the form of K80 and V100 cards, the number of cards is limited and the queueing time is long, especially for the V100 cards which can run faster simulations. This motivated our local user and us to apply for the present preparatory access project *2010PA5821*.

Classical molecular dynamics used in this workflow, to obtain accurate binding free energies with free energy perturbations, are performed with the AMBER 2020 [42] software which is known to scale well on HPC systems. This software requires a license and for the present study we use the site license from JSC. We have not performed any code optimisation.

For the present project, we requested 1000 node hours for 3 months and we were granted 25,000 core hours. Although the period was between 01.06.2021 and 30.11.2021, it actually started on 01.09.2021.

Performance comparison of an AMBER Free Energy Perturbation simulation running on our local cluster (LC) and on Juwels Boost (JB) cluster at JSC, Germany. For LC, a pure MPI simulation with 28 cores, one using 2 K80 GPUs, and one using 2 V100 GPUs were done. In JB, a simulation running with 48 MPIs and one using 4 A100 GPUs were performed. All simulations ran on a single node in all cases.

Architecture	Performance (ns/day)
28 MPI, LC	11
4 MPI, 2 K80, LC	38
2 MPI, 2 V100, LC	100
48 MPI, JB	18
4 MPI, 4 A100, JB	123

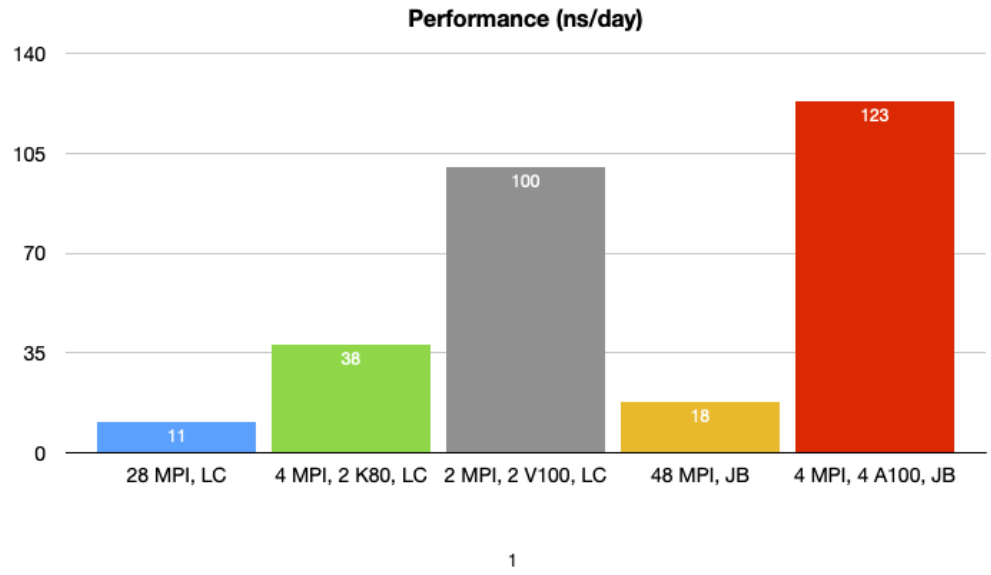


Figure 63: Comparison of the performance of AMBER software for a FEP simulation on our local (LC) cluster and on JUWELS Booster (JB).

For this FEP test case, the observed speedup (by running a single simulation) on JB was ~3.2x (123/38) w.r.t. the LC, using A100 and K80 GPUs on JB and LC, respectively, see Figure 63. The speedup is lower, 1.2x, if one uses the V100 GPUs on our LC. These speedups, for V100 GPUs, could be obtained in our LC in case the number of GPUs available were large enough. However, these resources are limited especially in the case of V100 cards which results in a long queueing time. The latter can be even longer than the actual simulation time. Thus, in practice it is more realistic to compare the speedup that can be obtained with the pure CPU simulation on the LC and the A100 on JB, which results in a 11.1x (123/11) performance gain.

Volume of core hours

The free energy perturbation workflow involves several independent "windows" for each protein+ligand system, in the present case 26 windows are required. Each window will be sampled for 10 ns. A total of 200 protein+ligand systems will be studied, where each system will involve 8 different sets of charges. In addition to this, 5 independent copies of each system will be run for the validation of the results. Thus, the total amount of ns for the whole project, upon applying for project access, is as follows:

200 (systems) x26 (windows) x8 (set of charges) x5 (validation) x10 (ns per simulation) = 2,080,000 ns (whole project)

The AMBER20 CPU implementation (baseline) can run 11 ns/day by using 28 cores, which means that for running 11 ns one should spend 28 cores*24 hours = 672 core hours, or in other words 61 core hours/ns. From the total number of ns computed above, the resulting amount of core-hours is 126,880,000 core-hours (2,080,000 ns x 61 core hours/ns).

Estimated saved core-hours

It is difficult to clearly define the volume of core.hours saved but the run on JB have a maximum speedup is 11.1x. In that perspective, running on JB will be way faster and that would enable substantial CPU hours gain on our local cluster for users who cannot benefit from the GPU acceleration

Outcome of the project

Based on the performance of the simulations observed on the A100 cards and the amount of core-hours that can be used in a Project Access project, the local user decided to apply for this type of project. We have submitted a proposal for Project Access for JUWELS Booster at JUELICH.

6 Summary

Four parallel tasks on application enabling and porting services in Work Package 7 of PRACE-6IP have been described including reports on the supported applications and activities. These two activities have been organised into support projects formed on a basis of either scaling and optimisation support for Preparatory Access, DECI, SHAPE and the HLSTs.

6.1 Applications Enabling Services for Preparatory Access

During PRACE-6IP Task 7.1 successfully performed ten cut-offs for preparatory access including the associated review process and support for the approved projects. In total 21 Preparatory Access projects have been supported by T7.1. Twelve projects were finally reported within this deliverable. In total seven new white papers, coving more details on the outcome of the PA projects were created in the frame of the project.

6.2 Applications Enabling Services for Industry

SHAPE has continued to help SMEs to try out HPC and there are many ways to see that this has been successful including the feedback from surveys. Proposal numbers have more than doubled for the last few calls with the number of countries participating having increased. This has been due to a combination of effort on the part of those working directly on the SHAPE programme, the input from Wahid Rofagha, the PRACE Industry Liaison Officer, and the implementation of the SHAPE+ programme. White Papers and Success Stories also show good work having been carried out with a number of SMEs. We look forward to working alongside and continuing to share experiences with the EuroHPC Competence Centres.

6.3 DECI Management and Applications Porting

DECI continues to be very popular with researchers across Europe with large numbers of proposals received for each call across a number of countries. Support for projects will continue until the end of PRACE-6IP and we look forward to discussions with EuroHPC about how DECI activities may continue over the next few years.

6.4 Enhancing the High Level Support teams

The task aimed at enabling users and communities to large-scale architectures such as Tier-0 systems and pre-exascale systems.

During this project, the team work on five different applications:

- NEMO
- NAMD
- AMBER
- VASP
- Full Wavefield Migration application

Most of the applications were ported and optimised regarding specific needs of users who are intensively using HPC resources. But they are also community applications. In that sense, the

work performed within this task will not only benefit to the users we supported, but also to the whole scientific community using those applications.

The work performed enabled applications to use leading edge CPU architectures (such as the new AMD CPUs that are more and more present on the European HPC ecosystem) as well as leading edge GPU architecture (such as the Tier-0 JUWELS Booster, involving the latest GPU available in the landscape at the time this report is written, the NVIDIA A100).

Thanks to the dedication of the team to support the users, the scientific communities moved to Tier-0 systems, using the preparatory access protocol, and even using the standard regular call for some projects at the end of this activity.

While there was not a lot of partners and PMs involved in this task, we managed to show the benefit of such activity as the estimated amount of CPU hours saved (hundreds of millions) is substantial thanks to the work performed by the team.